

### III. INNOWACJE PEDAGOGICZNE

STANISŁAW UBERMANOWICZ  
Uniwersytet im. Adama Mickiewicza  
w Poznaniu

HANNA BIELAWSKA  
Szkoła Podstawowa nr 2  
w Gnieźnie

#### CZYNNOŚCIOWE KSZTAŁTOWANIE POJĘĆ

ABSTRACT. Ubermanowicz Stanisław, Bielawska Hanna, *Czynnościowe kształtowanie pojęć* [Functional shaping of concepts]. „Neodidagmata” 25/26, Poznań 2003, Adam Mickiewicz University Press, pp. 147-166. ISBN 83-232-1366-6. ISSN 0077-653X.

The article describes a method of providing a school lesson with concept shaping in the form of practical acting. The subject matter comprises such terminology of computer science as: organization of information, concatenation of files, hierarchy of tasks, structuring of data, visualizing of data, conversion of formats, algorithmization of problems, and implementation of ideas. Concept defining is compared with operational defining. A functional procedure of interiorization of concepts is offered.

*Stanisław Ubermanowicz, Zakład Technologii Kształcenia, Wydział Studiów Edukacyjnych, Uniwersytet im. Adam Mickiewicza, ul. Słowackiego 20, 60-823 Poznań, Polska-Poland; e-mail: uberman@amu.edu.pl*

*Hanna Bielawska, Szkoła Podstawowa nr 2 im. Księdza Majora Mateusza Zabłockiego, ul. Lubieńskiego 7, 62-200 Gniezno, Polska-Poland; e-mail: habiel@poczta.onet.pl*

Ekspansja komputeryzacji, dynamiczne rozprzestrzenianie się zastosowań informatyki, zwiększenie roli komunikacji i wymiany informacji – wszystko to miało wpływ na pojawienie się nowej dziedziny – *technologii informacyjnej*. Ze względu na znaczną przydatność dziedzinę tę wprowadzono do szkół ponadgimnazjalnych jako odrębny przedmiot nauczania. Już wcześniej zobligowano natomiast nauczycieli innych przedmiotów do stosowania nowych technologii na swoich zajęciach.

Choć wydaje się dziwne, to jednak oczywista staje się konieczność stosowania technologii informacyjnych także na przedmiocie *informatyka*, realizowanym na każdym etapie kształcenia. Przedstawione tu propozycje odnoszą się do takiej właśnie sytuacji, tj. wykorzystywania na lekcjach informatyki komputera nie tylko jako obiektu działań lub jako środka dydaktycznego, lecz przede wszystkim jako me-

dium używanego w funkcji środka-metody kształcenia i samokształcenia obudowanego kompleksowo za pomocą starannie dobranych programów, systemów, metod i technik informacyjnych.

Wyodrębniając nową dziedzinę, informatycy dążyli do tego, ażeby ich pierwotna dziedzina była kojarzona głównie z dyscypliną naukową, zajmującą się teorią i praktyką przetwarzania informacji. Informatyka – jak każda nauka – opiera się na ściśle zdefiniowanych pojęciach, kluczowych dla danego obszaru wiedzy. Wyraźnie rozróżnia się przy tym uogólniające definicje naukowe od rozumienia potocznego, przy czym praktykowane w szkołach wyuczenie się definicji na pamięć nie jest równoznaczne ze zrozumieniem pojęcia. Często utrwalana jest wręcz błędna konotacja pojęć, np. edycja automatycznie kojarzy się z przetwarzaniem tekstu przy użyciu konkretnej wersji edytora, a tymczasem może to być także edycja dźwięku, obrazów, filmów, programów radiowych i telewizyjnych.

Kształcenie informatyczne w szkole nie powinno sprowadzać się jedynie do nauczania obsługi komputera bądź nabywania umiejętności użytkowania programów. Obok ćwiczenia umiejętności praktycznych przedmiot szkolny związany z dziedziną naukową powinien przekazywać także obszerny zasób wiedzy pojęciowej, będącej dorobkiem danej dyscypliny. Uczniowie nie powinni jednak uczyć się pojęć na zasadzie wchłaniania, reprodukcji i odtwarzania. Skuteczniejszą formą uczenia się jest działanie praktyczne. Ta czynność, którą uczeń wykonał, lepiej utrwała się w pamięci, a ponadto samodzielne wykonanie zadania ułatwia zrozumienie istoty danego pojęcia. W takiej sytuacji zachodzi proces przyswajania czynności z budowaniem struktur śladów i wzorców w umyśle.

**Pojęcie** jest umysłową abstrakcją, symbolem bądź odwzorowaniem obiektu, cechy, zjawiska lub relacji. Jest więc jedynie reprezentacją rzeczywistości powstałą w umyśle człowieka. W procesach komunikowania społecznego do opisywania pojęć służą definicje pojęciowe lub operacyjne. Istotą tych pierwszych jest odwzorowywanie za pomocą innych, już wcześniej przyswojonych pojęć. Zwróćmy jednak uwagę na trudność opisywania za pomocą innych słów, zwłaszcza gdy dane słowo ma wyspecjalizowane znaczenie. Lepiej jest w takiej sytuacji posłużyć się definiowaniem operacyjnym, łączącym poziom pojęciowo-teoretyczny z empiryczno-obszernym. Wymaga to niekiedy przekształcenia pojęć określających obiekty (np. „struktura”) w pojęcia czynnościowe („strukturyzacja”), które łatwiej jest uczniom przyswoić poprzez osobiste doświadczenie definiowanego procesu.

W niniejszym opracowaniu przedstawiamy przykłady realizacji zajęć szkolnych z zastosowaniem czynnościowej metody kształtowania pojęć. Przyjęliśmy jednolitą konwencję umieszczania przy każdym z zagadnień najpierw definicji pojęciowych, a następnie opisów operacyjnych. Te ostatnie, zawierające zadania dla uczniów i komentarze nauczyciela, z konieczności mogły być tylko zasygnalizowane, gdyż w pełnej wersji metodycznej stanowią: zbiór szczegółowych procedur postępowania, listę pytań naprowadzających, treść zadań do samodzielnego wykonania, sugestię prawidłowych rozwiązań, wypunktowanie charakterystycznych

momentów w danej czynności i niuansów, jakie powinien zaakcentować nauczyciel. Tak obszernie treści i szczegółowe procedury opisać można jedynie w formie publikacji podręcznikowej.

Z bogatego zasobu zagadnień przedmiotowych informatyki wybraliśmy pojęcia przykładowe, szczególnie istotne dla tej dziedziny, lecz nie mniej ważne także w szerszym aspekcie uporządkowanego pojmowania świata. Takie zagadnienia, jak: *organizacja informacji, scalanie zbiorów, hierarchizacja zadań, strukturyzacja danych, wizualizacja danych, konwersja formatów, algorytmizacja problemu oraz implementacja pomysłu* występują w treściach programowych na różnych poziomach kształcenia. Warto zwrócić uwagę na zastosowany tu sposób operacjonalizacji pojęć oraz na powiązanie procesów z obiektami, co znajduje odbicie w jednolitych śródtytułach, mogących stanowić tematy kolejnych lekcji\*.

## ORGANIZACJA INFORMACJI

Informacja, pomimo że jest obiektem abstrakcyjnym, może być kodowana w postaci danych, które organizowane są w taki sposób, ażeby można było je najefektywniej pozyskiwać, przechowywać, przesyłać, przetwarzać i udostępniać. W czynnościach wprowadzania danych do komputera oraz ich wyświetlania dużą rolę odgrywa interfejs użytkownika. Jest to przemyślany sposób komunikowania się człowieka z komputerem, polegający na wyświetlaniu okien dialogowych z komunikatami, z polami przeznaczonymi do wpisywania i z przyciskami służącymi do zatwierdzania. Od zorganizowania interfejsu i od postaci informacji na ekranie zależy łatwość posługiwania się danym programem.

### Przykład 1. Kalendarz osobisty

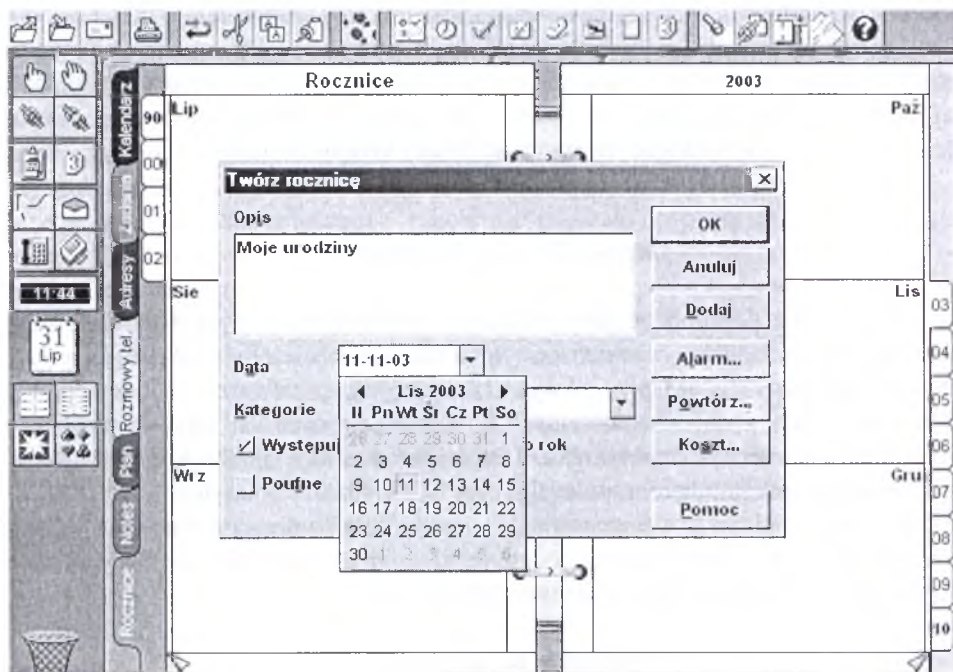
Zadaniem uczniów jest rozpoznanie mechanizmów organizowania informacji w wirtualnej postaci graficznej, imitującej naturalny kołonoratnik-kalendarz. Do kształtowania tego pojęcia najlepiej jest wykorzystać program Lotus Organizer 2.1, będący przyjaznym, atrakcyjnym i chętnie poznawanym przez uczniów programem typu PIM (*Personal Information Manager*). Zajęcia te można realizować na dowolnym poziomie nauczania.

#### *Czynności ucznia*

Każdy z uczniów wprowadza po dwie ważne dla niego rocznice, np. datę swoich urodzin oraz imienin. Następnie na Planie rocznym zaznacza daty związane z dwoma wolnymi od zajęć dniami świątecznymi. Kolejną czynnością jest wydanie polecenia reorganizacji informacji, w wyniku czego następuje automatyczne rozdystrybucowanie wcześniej wpisanych rocznic oraz dni wolnych na odpowiednich stro-

\* Pliki do obudowy zajęć można pobrać z [www.wse.amu.edu.pl/neo/](http://www.wse.amu.edu.pl/neo/)

nicach Kalendarza. Uczeń sprawdza, czy w Kalendarzu, w dniu dla niego odświetnym, pojawiają się w kolejnych latach informacje o tej rocznicy. W dalszej fazie realizacji zadania należy w Kalendarzu zaplanować po trzy dowolne czynności na bieżący dzień, na ściśle określonej godzinie, w taki sposób, ażeby czas dwóch zdarzeń częściowo się pokrywał. Program wykryje konflikt zdarzeń, informując użytkownika o niepoprawnym organizowaniu swego czasu.



Ryc. 1. Organizacja informacji w kołonotatniku-kalendarzu PIM

### *Komentarz nauczyciela*

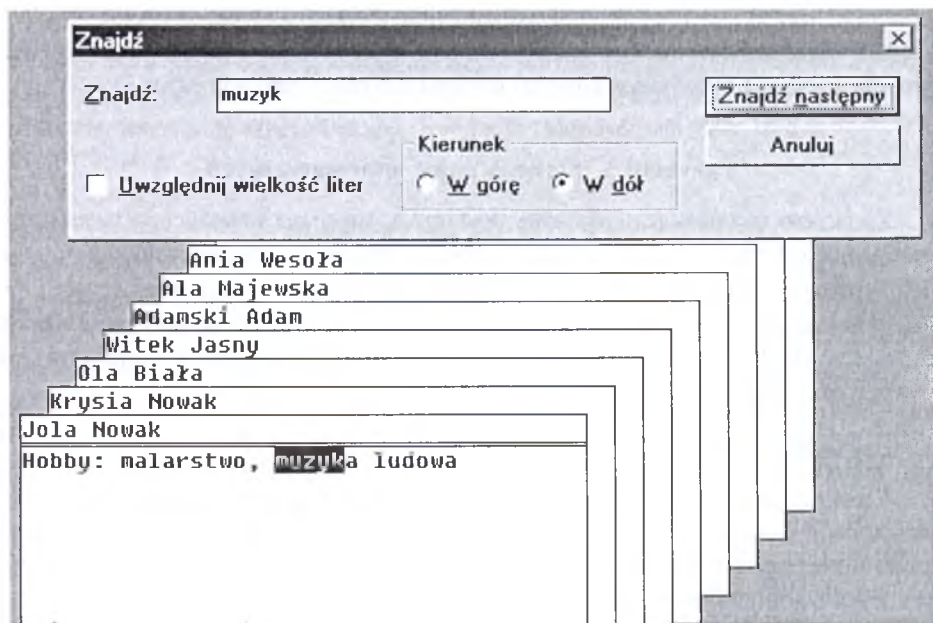
Głównym celem zajęć jest poznanie różnych, atrakcyjnych graficznie form organizowania informacji. Dodatkowo występuje tu pojęcie organizowania własnych zajęć, gdyż program PIM służy do bezkonfliktowego planowania czynności codziennych i długoterminowych. Wewnętrzne mechanizmy aplikacji same organizują, nadzorują i przenoszą informacje pomiędzy różnymi sekcjami kołonotatnika-kalendarza. Nauczyciel zwraca uwagę na wirtualną formę Organizera, polegającą na możliwości zmiany postaci wyświetlania informacji oraz na takich modyfikacjach, które nie są możliwe w kalendarzu drukowanym. Podkreśla przy tym ważną rolę intuicyjnego, łatwego w obsłudze interfejsu użytkownika.

## SCALANIE ZBIORÓW

Coraz częściej w realizacji większych projektów informatycznych uczestniczy grupa osób. Każdy z wykonawców odpowiedzialny jest za realizację wydzielonych fragmentów, które w końcowej fazie należy scalić. Proces scalania rozproszonych zasobów informacji zapisanych w zgodnym formacie zwany jest przez informatyków *konkatenacją*. Warto na lekcjach wykonywać projekty zespołowe i sumować produkty cząstkowe w finalną całość. Celem obu proponowanych zadań jest scalanie porcji danych – w jednym przykładzie uczniowie przenoszą pliki na dyskietkach, a w drugim przesyłają za pomocą lokalnej sieci komputerowej.

## Przykład 2. Kartoteka – moje hobby

Zadanie każdego z uczniów polega na wypełnieniu elektronicznego formularza z elementarnymi danymi (nazwisko, imię i hobby), a następnie na scaleniu informacji dotyczących całej grupy, zarchiwizowaniu bazy danych i wyszukaniu osób o podobnych zainteresowaniach. Do kształtowania u uczniów pojęcia *scalania zbiorów danych* warto wykorzystać systemową aplikację Kartoteka (z Windows 95). Zajęcia można przeprowadzić już na poziomie szkoły podstawowej.



Ryc. 2. Fragment scalonej Kartoteki z uruchomioną funkcją szukania

### *Czynności ucznia*

Pierwszy z uczniów pracujący przy komputerze wpisuje w wierszu indeksowym swoje nazwisko i imię, a następnie w polu opisowym fiszki, w sposób jak najbardziej zwięzły wymienia swoje hobby. Po wypełnieniu fiszki kartotecznej zapisuje dane do pliku. Następnie identyczne informacje wprowadza drugi z uczniów, wybierając funkcję *Dołącz*. Po zapisaniu danych na twardym dysku, umieszcza duplikat pliku na dyskietce, którą przekazuje na sąsiedni stolik. Dane otrzymane na dyskietkach od innych uczniów integruje z własną kartoteką za pomocą funkcji *Łącz*. Uczniowie powtarzają cykl scalania wszystkich fiszek, podając sobie dyskietki zawsze w tę samą stronę. Po zapisaniu na dysku pliku Kartoteki z kompletem danych, uczniowie przystępują do odszukania, czy w klasie ktoś inny ma podobne zainteresowania.

### *Komentarz nauczyciela*

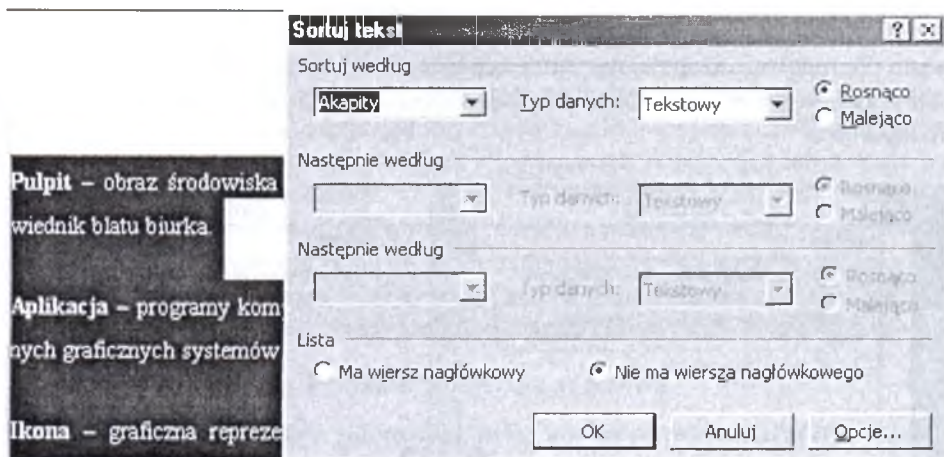
Scalanie jest ważne przy wprowadzaniu obszernych zbiorów danych, gdyż rozłożenie czynności na zespół może skrócić czas pracy niezbędny do wykonania wspólnego projektu. Przy okazji uczniowie ćwiczą umiejętność posługiwania się zewnętrznymi nośnikami informacji. Nauczyciel zwraca uwagę na różnice między funkcjami *Dołącz* i *Łącz*. Ponadto akcentuje odróżnianie pliku źródłowego z danymi cząstkowymi od pliku docelowego z danymi scalonymi. Przypomina też o konieczności zamykania programu przed wyjęciem dyskietki. Podczas przeszukiwania zaleca skracanie słów kluczowych w taki sposób, aby nie zawierały końcówek fleksyjnych, a jedynie były początkowymi literami lub rdzeniami wyrazów.

### **Przykład 3. Słownik pojęć informatycznych**

Zadaniem uczniów jest wpisanie dostarczonych przez nauczyciela haseł z obszaru pojęć informatycznych wraz z ich objaśnieniami, a następnie scalenie wszystkich zebranych w klasie dokumentów w jeden alfabetycznie uporządkowany Słownik. Do kształtowania pojęcia *scalania dokumentów* należy wykorzystać edytor tekstu, np. MS Word, przy czym do wymiany dokumentów warto wykorzystać sieć wewnętrzną. Zajęcia można realizować już w szkole podstawowej.

### *Czynności ucznia*

Każdy z uczniów wpisuje otrzymane od nauczyciela dwa różne hasła. Następnie zapisuje dokument z hasłami do jednego pliku, który będzie traktowany jako ogólnodostępny w sieci tekst źródłowy, oraz do drugiego, który będzie dokumentem zbiorczym, otwartym do wyłącznego użytku. Za pomocą funkcji *Wstaw → Plik...* z otoczenia sieciowego wybiera kolejne nazwy komputerów tyle razy, aż dołączy wszystkie pliki źródłowe napisane przez innych uczniów. Następnie sortuje hasła według porządku alfabetycznego.



Ryc. 3. Fragment scalonego Słownika z uruchomioną funkcją sortowania

### Komentarz nauczyciela

Wykonana czynność scalania plików pozwala w krótkim czasie napisać obszerny dokument. Nauczyciel powinien zwrócić uwagę na to, by objaśnienie dotyczące jednego hasła było napisane w jednym akapicie, w przeciwnym wypadku uczniowie będą mieli problemy podczas sortowania. Przed wstawieniem do swojego dokumentu treści z innego pliku, należy wskazać kursorem miejsce wstawienia. Warto przy okazji kształtować u uczniów nawyk systematycznej archiwizacji na dysku, aby nie stracić efektów pracy w wypadku zawieszenia się systemu.

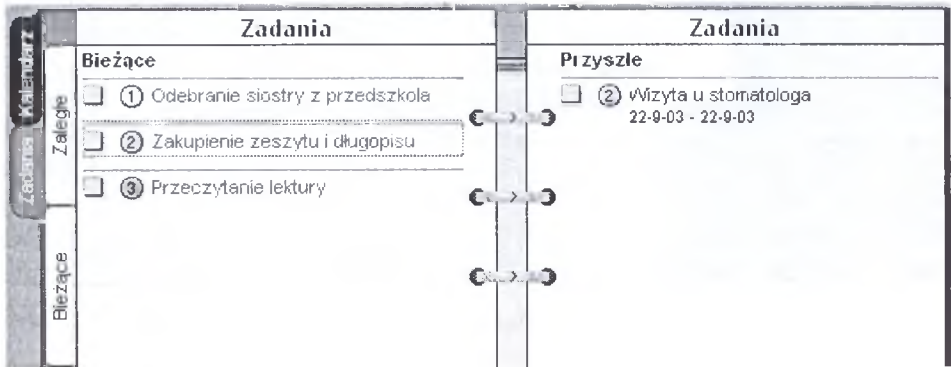
## HIERARCHIZACJA ZADAŃ

Hierarchia oznacza uporządkowany układ elementów w relacji wyższości ↔ niższości według ściśle zdefiniowanego kryterium. Najczęściej stosowanym kryterium jest porządkowanie *od ogółu do szczegółu*. Przy kształtowaniu tego pojęcia uczniowie poznają i stosują także inne kryteria hierarchizacji, takie jak: *pierwszoplanowość zadań* oraz *następstwo w czasie*. Podczas zajęć uruchamiają mechanizmy porządkowania zadań według priorytetów, chronologii i stopnia realizacji oraz nabywają umiejętności grupowania według kategorii.

### Przykład 4. Moje zajęcia

Zadaniem uczniów jest zaplanowanie swoich czynności, z ustaleniem dla wszystkich tych zajęć kryterium ważności oraz pilności. Polega to na wprowadzeniu zwartych opisów czynności do programu typu PIM. Zadaniom terminowym

przypisywana jest także przewidywana data wykonania. Dodatkowo czynności warto pogrupować, uzupełniając istniejące kategorie. Do tego celu najbardziej przydatny jest program Lotus Organizer 2.1. Zajęcia można realizować już w szkole podstawowej.



Ryc. 4. Hierarchizacja zadań w kalendarzu-kalendarzu PIM

### *Czynności ucznia*

Każdy z uczniów wprowadza do elektronicznego kalendarza-kalendarza przynajmniej po cztery zadania, które planuje wykonać w najbliższym czasie. Czynnościom tym nadaje priorytety 1, 2 lub 3, zależne od pilności danego zajęcia. Planowane zadania stara się pogrupować według zawartych w programie kategorii, a jeśli one nie pasują, to dopisuje nową kategorię. Dla wybranych czynności określa daty wykonania – wcześniejsze bądź późniejsze. Po zaplanowaniu zadań archiwizuje informacje na dysku, po czym zmienia sposób wyświetlania zadań według różnego porządku: priorytetu, statusu, daty rozpoczęcia oraz kategorii.

### *Komentarz nauczyciela*

Nauczyciel zwraca uwagę na możliwość wyświetlania informacji o planowanych zajęciach w różnych formach. Pierwszą formą jest porządkowanie według priorytetu, co kształtuje pojęcie hierarchizacji w aspekcie ważności zadań. Drugą formą jest grupowanie według statusu zadań: *Zaległe*, *Bieżące*, *Przyszłe* i *Wykonane*, co ułatwia śledzenie systematyczności w realizacji własnych planów. Trzecią formą jest ułożenie zadań według dat: rozpoczęcia, zakończenia lub częściowego wykonania, co kształtuje pojęcie hierarchii chronologicznej i pojęcie terminowości. Czwartą formą jest grupowanie według kategorii, np.: *Szkoła*, *Dom* itp., co może służyć dodatkowo do kształtowania pojęcia zbiorów.

## STRUKTURYZACJA DANYCH

Strukturyzacja jest czynnością polegającą na budowaniu zbiorów danych w postaci jednorodnych konstrukcji, jednoznacznie zdefiniowanych oczekiwaną zawartością treściową oraz uporządkowanych i powiązanych relacjami według ściśle wyznaczonych kryteriów. Każdemu z elementów struktury przypisuje się charakterystyczną nazwę ogólną, określającą rodzaj obiektu, a następnie nadaje nazwę własną, konieczną przy odwoływaniu się do danego obiektu. Najważniejszym zagadnieniem w strukturyzacji danych jest zdecydowanie o tym, jakie pola ma zawierać pojedynczy rekord, a więc – które z informacji dotyczących danej osoby są niezbędne.

### Przykład 5. Książka adresowa

Zadanie jest projektem zbiorowym, polegającym na tworzeniu adresowej bazy danych uczniów. Uczniowie projektują optymalną i modyfikują istniejącą strukturę pól w elektronicznej książce adresowej, po czym wprowadzają swoje dane osobowe oraz dane kontaktowe jednego z rodziców. Do zajęć warto wykorzystać program Lotus Organizer 2.1. Temat strukturyzacji zalecamy do realizacji w gimnazjum, łącznie z omawianiem pojęcia *selekcji informacji*.

Furmaniak	
	P   D
<b>Furmaniak, Monika</b>	
Monika Furmaniak	
Słowackiego 2/3	
Gniezno, 62-200	
Data ur.	9.04.1989
Miejsce ur.	Gniezno
Tel. 1	0-61 4182234
Tel. 2	
Fax	
Email	
Imię ojca	Adam
Imię matki	Ewa
Kategorie	

Ryc. 5. Struktura Karty z danymi ucznia

### Czynności ucznia

Pierwszą fazą projektowania struktury bazy danych jest metoda „burzy mózgów”, podczas której uczniowie samodzielnie próbują określić, jakie najważniejsze informacje powinna zawierać szkolna książka adresowa. Zwykle wymieniają: imię, nazwisko, adres, numer telefonu, datę i miejsce urodzenia, imię ojca, imię matki oraz informacje o miejscu pracy przynajmniej jednego rodzica (lub opiekuna) w celu

kontaktu. Następnie dopasowują istniejącą strukturę sekcji Adresy Organizera, zmieniając nazwy pól tak, aby były zgodne z przygotowanym projektem bazy danych. W dalszej fazie zajęć wypełniają formularze danych i przeglądają zawartość kart. Na zakończenie wybierają przyciski zmieniające ilość wyświetlanych informacji, stosownie do opcji: *wszystko*, *adres*, *kontakt* lub *telefon*.

### *Komentarz nauczyciela*

Nauczyciel zwraca uwagę na strukturę bazy danych, którą tutaj tworzą: rekordy, pola, indeksy, karty (raporty) i zakładki grupujące rekordy. Podkreśla zwłaszcza różnicę między nazwą pola tekstowego a jego tytułem. Szczególną rolę pełni pole indeksowe, w tym wypadku jest to nazwisko ucznia. Według tego pola sortowane są alfabetycznie rekordy bazy danych, ponadto tworzone są relacje pomiędzy kartą z danymi ucznia a kartą z danymi jego rodziców.

W wypadku większych zbiorów baz danych ważnym zagadnieniem jest *selekcja informacji*. Nauczyciel akcentuje to, że dostępne są różne formy selekcji i wyświetlania tylko fragmentów danych. Pierwsza forma polega na wyborze kart uczniów za pomocą zakładek grupujących alfabetycznie. Druga forma selekcji informacji dotyczy przełączania pomiędzy kartami z danymi uczniów a kartami rodziców. Trzecia umożliwia czterostopniową redukcję wyświetlanych informacji, od przeznaczenia całej strony na jednego ucznia aż do niezbędnego minimum mieszczącego się w jednym wierszu danych.

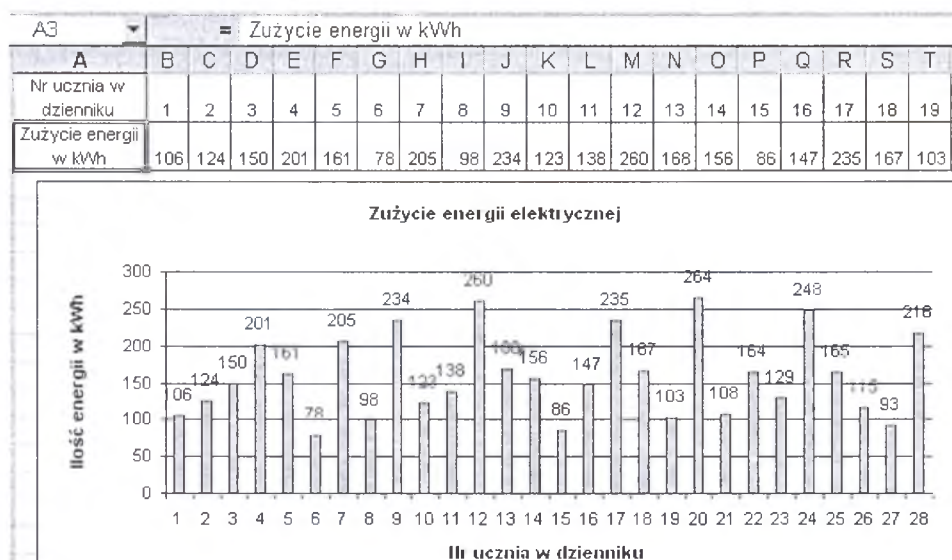
## WIZUALIZACJA DANYCH

Poprzez wizualizację rozumiemy wszelkie formy obrazowania danych, obiektów, modeli teoretycznych, procesów itd. Z pojęciem tym uczniowie spotykają się na każdej lekcji informatyki, często nie zdając sobie z tego sprawy: w środowisku Windows patrzą na *Pulpit*, na którym poukładane są ikony, czyli graficzne reprezentacje aplikacji; na *Pulpicie* znajduje się *Pasek zadań* z przyciskiem *Start* itd. Wirtualne przyciski, jak np. *Zamknij*, *Przywróć*, *OK*, *Anuluj*, często pojawiają się w filozofii interfejsu okienkowego. Nie są to jednak fizyczne przyciski, lecz ich wizualizacja, często wzmocniana reprezentacją ikonyczną. W wizualizacji ważnym zagadnieniem jest też ożywianie obrazów – animacja. Ze względu na rosnącą rolę wizualizacji we wszystkich dziedzinach życia ważne jest wyrabianie u uczniów nawyku ilustrowania właśnie za pomocą technologii informacyjnych.

### **Przykład 6. Zużycie energii. Co pijemy na śniadanie?**

Zadaniem uczniów jest wprowadzenie do arkusza kalkulacyjnego Excel wyników dwóch przeprowadzonych badań ilościowych i wygenerowanie wykresów optymalnie dopasowanych do charakteru danych. Analiza danych statystycznych.

wyników ankiet itp. staje się łatwiejsza, gdy posłużymy się graficzną reprezentacją danych liczbowych. Zajęcia takie można realizować już w szkole podstawowej.



Ryc. 6. Wizualizacja danych empirycznych dotyczących zużycia energii

### Czynności ucznia

Uczniowie wcześniej zbierają informacje na temat ilości zużytej energii elektrycznej w gospodarstwach domowych na podstawie przyniesionych rachunków. W arkuszu kalkulacyjnym każdy z uczniów przygotowuje tabelę miesięcznego zużycia prądu w gospodarstwach uczniów z danej klasy. Zaznacza właściwy obszar komórek i wizualizuje zgromadzone dane w odpowiednio dopasowanej postaci graficznej, wykorzystując do tego celu Kreator wykresów.

W tym samym skrócie uczeń wykonuje drugą część zadania. Przygotowuje tabelę przedstawiającą fragment wyników przeprowadzonej w szkole ankiety na temat „Zdrowy tryb życia”. Następnie, korzystając z kreatora wykresów, generuje diagram kołowy, ilustrujący kwestię np. tego, co uczniowie najczęściej piją podczas śniadania: mleko, herbatę, kakao, kawę, sok czy inny napój.

### Komentarz nauczyciela

Nauczyciel powinien zwrócić uwagę uczniów na to, iż rodzaj wykresu musi być optymalnie dopasowany do rodzaju danych. Przygotowując w arkuszu kalkulacyjnym tabelę i na tej podstawie generując wykres, uczniowie mają możliwość obserwowania różnic na diagramach inicjowanych przez zróżnicowane dane. Zmiany wysokości słupków oraz zmiany proporcji wycinków koła są znacznie bar-

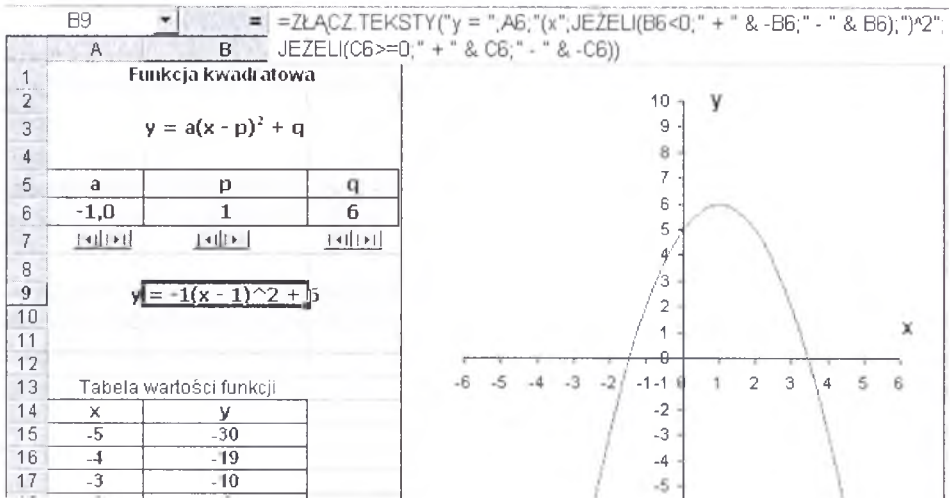
dziej wyraziste niż zmiany liczb w tabelach. Analizując wyniki przeprowadzonych badań, znacznie łatwiej można wyciągać odpowiednie wnioski i porównywać wyniki, jeśli zamiast tabel posłużymy się graficzną interpretacją danych. Zastosowanie arkusza kalkulacyjnego znacznie skraca czas wykonania rysunku w porównaniu z rysowaniem odręcznym, a uczeń ma okazję samodzielnie wykonać atrakcyjne graficznie wizualizacje.

### Przykład 7. Funkcja kwadratowa

Kolejny projekt oparty jest na połączeniu wizualizacji z animacją. Uczniowie projektują i wykonują elektroniczny arkusz, generujący wykres funkcji kwadratowej podanej w postaci kanonicznej:  $y = a(x - p)^2 + q$ ,  $x \in R$ ,  $a \neq 0$ . Do realizacji wykorzystuje się np. program MS Excel. Ze względu na stopień skomplikowania projekt zalecany jest do realizacji na zajęciach z informatyki w szkołach ponadgimnazjalnych.

#### Czynności ucznia

Wspólnie z nauczycielem ustalane są teksty i formuły obliczeniowe wpisywane do odpowiednich komórek arkusza. W projekcie niezbędne jest zastosowanie zaawansowanych formuł operowania nie tylko na liczbach, lecz także na łańcuchach tekstowych. Ponadto konieczne jest użycie elementów programowania obiektowego.



Ryc. 7. Wizualizacja zmienności funkcji z możliwością animacji

Do zmian wartości posłuży utworzony za pomocą Przybornika formantów obiekt, nazwany niefortunnie przez Microsoft *przyciskiem pokrętła*(!). Uczniowie definiują właściwości tego formantu, wiążąc zakres wartości zmian ze zmiennością

argumentu funkcji. Do liczbowego przedstawienia przebiegu zmienności funkcji niezbędne jest sporządzenie tabelki z iteracyjną serią argumentów i adekwatną serią wyników. Przy generowaniu rysunku uczniowie wykorzystują Kreator wykresów, wybierając typ *XY Punktowy* oraz podtyp *Wykres punktowy z punktami danych połączonymi wygładzonymi liniami bez znaczników*.

### *Komentarz nauczyciela*

Nauczyciel organizuje pracę uczniów, wydając zalecenia i omawiając z nimi optymalny sposób rozwiązania zadania. Podczas projektowania zwraca uwagę na informatyczne aspekty zadania: sposób adresowania komórek, bezbłądność syntaktyczną definiowanych formuł oraz na zoptymalizowane (a nie automatyczne) skalowanie osi wykresu. Przygotowany przez uczniów projekt może być wykorzystany na lekcjach matematyki do obserwacji, jak zmieniają się własności funkcji kwadratowej w zależności od zmiany współczynników.

## KONWERSJA FORMATÓW

Przetwarzanie informacji wymaga standaryzacji postaci, jaką tym informacjom nadajemy. Jednak, ze względu na ciągłe doskonalenie technologii informacyjnych, trudno jest uzgodnić uniwersalne i niezmiennie formaty danych. Tymczasem użytkownik musi mieć możliwość: przenoszenia obiektów pomiędzy aplikacjami, skalania różnych postaci informacji, prezentacji projektów w różnych miejscach, kondensowania danych przy archiwizacji, kompresji w celu szybkiej transmisji itd. Niezbędna staje się *konwersja formatów*, a więc operacja polegająca na zmianie jednego standardu w inny. Do formatowania informacji służą edytory, które posiadają wewnętrzne moduły konwersji formatów, a ponadto dodatkowo instalowane w zależności od potrzeb zewnętrzne biblioteki rozszerzające konwersję. Konwersja zaawansowanych technologicznie wytworów informatycznych nie zawsze jest możliwa, a w wypadku przetwarzania na uproszczony lub silnie skompresowany format może prowadzić do utraty istotnych właściwości oryginału lub do pogorszenia jakości konwertowanego duplikatu.

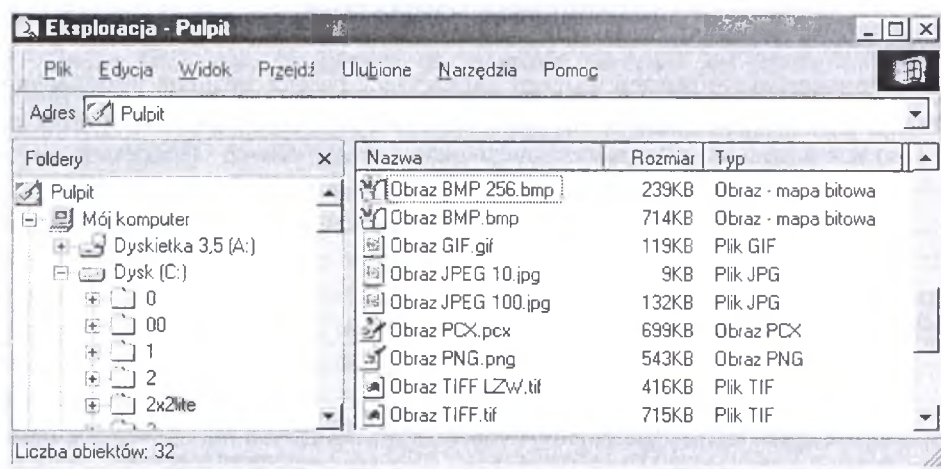
### **Przykład 8. Obraz z martwą naturą**

Zadaniem uczniów jest przekonwertowanie oryginalnego obrazu o wysokiej rozdzielczości i profesjonalnej jakości, zapisanego pierwotnie w formacie mapy bitowej na inne formaty, z zachowaniem takiej samej wysokości i szerokości, lecz połączone ze zmianą liczby definiowanych kolorów oraz zmianą głębokości kompresji kosztem jakości. Podczas kształtowania pojęcia *konwersji formatów* szczególnie spektakularne jest przetwarzanie plików graficznych. Do przeprowadzenia

ćwiczenia warto wykorzystać program Microsoft Photo Editor zawarty w pakiecie MS Office. Zadanie to jest możliwe do realizacji na każdym poziomie nauczania.

### *Czynności ucznia*

Ćwiczący samodzielnie dokonują konwersji, każdorazowo z oryginalnego obrazu zdefiniowanego w grafice 24-bitowej True Color w formacie BMP, na osiem innych formatów: BMP 8-bitowe z paletą 256 kolorów, Graphics Interchange Format (GIF 256 kolorów), JPEG ze współczynnikiem jakości 90% (większy plik/dobra jakość), JPEG ze współczynnikiem jakości 10% (mniejszy plik/nizsza jakość), TIFF bez kompresji, TIFF z kompresją LZW, Portable Network Graphics (PNG) oraz PC Paintbrush (PCX). Uczniowie wnikliwie przyglądają się obrazom, a zwłaszcza odwzorowaniu szczegółów, i notują spostrzeżenia o jakości każdego z formatów oraz rozdzielczości wyrażonej w pikselach na centymetr. Po zakończeniu tworzenia plików uczniowie wykorzystując swoje notatki i odczytując za pomocą Eksploratora rozmiary plików w kilobajtach, przeprowadzają wspólnie analizę porównawczą i wymieniają się spostrzeżeniami.



Ryc. 8. Rozmiary plików graficznych w zależności od formatów i kompresji

### *Komentarz nauczyciela*

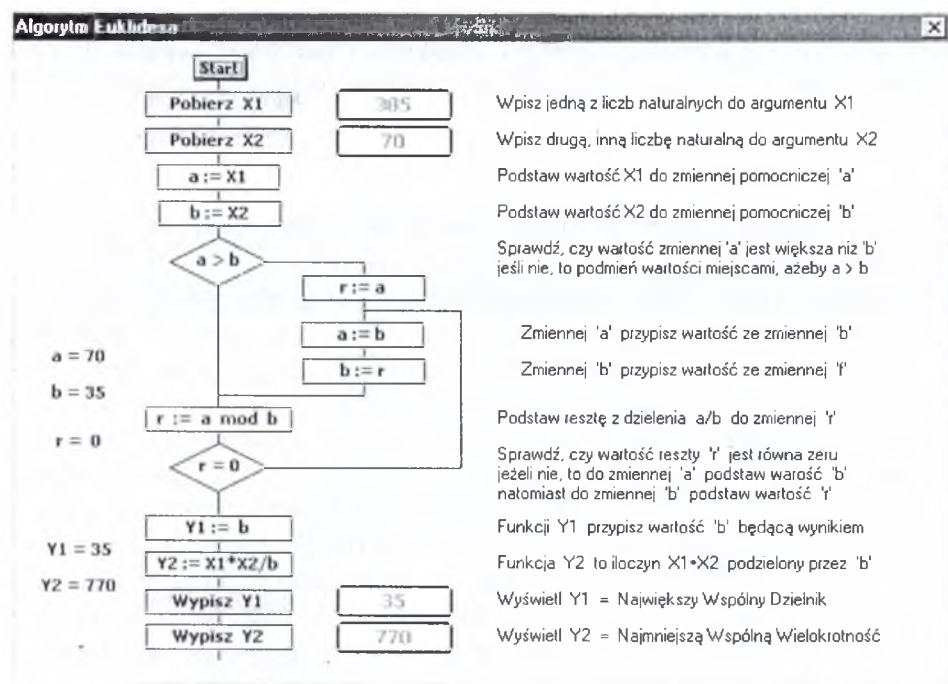
Należy zwrócić uwagę na to, że konwersja obrazu na różne formaty umożliwia porównanie jakości i ocenę przydatności poszczególnych sposobów kodowania. Nauczyciel zaleca uczniom zwłaszcza włączanie powiększonego wyświetlania obrazów (np. 200%) po to, ażeby lepiej dostrzegali powstające zniekształcenia spowodowane konwersją. Akcentuje takie niekorzystne zjawiska, jak: rozmycie szczegółów, zmniejszenie gradacji, rozlewanie się kształtów, zacieranie konturów, gubienie światłocieni, zjawisko mozaiki itp. Nauczyciel dodatkowo objaśnia wady i zalety poszczególnych formatów graficznych oraz ich optymalne zastosowania.

## ALGORYTMIZACJA PROBLEMU

Algorytm, to oparty na efektywnej koncepcji szczegółowy przepis na rozwiązanie danego problemu metodą operacji elementarnych. Często przedstawiany jest graficznie z uwagi na rozgałęzioną strukturę ścieżek przejść. Istotne jest, aby optymalną drogą skutecznie prowadził do wytyczonego celu. Kształtując u uczniów pojęcie algorytmizacji problemu, warto wykorzystać aplikację Algorytm Euklidesa. Program ten wizualizuje sposób znajdowania największego wspólnego dzielnika dwóch liczb poprzez kolejne dzielenia z szukaniem ostatniej reszty niezerowej. Uczniowie w liceach poznają wcześniej zagadnienie algorytmu Euklidesa na matematyce, w dziale poświęconym teorii liczb i wielomianom.

## Przykład 9. Największy wspólny dzielnik

Zadanie zalecane jest do przeprowadzenia w szkole ponadgimnazjalnej na zajęciach z informatyki, na początku, kiedy rozpoczyna się realizację bloku tematyki związanej z algorytmizacją. Korzystając z programu Algorytm Euklidesa (plik nwdw.exe), nauczyciel wyświetla krokowo algorytm w postaci schematu blokowego, a uczniowie odczytują znaczenie kolejnych kroków i starają się wyprzedzająco uzasadniać celowość ich wykonywania, zanim jeszcze zobaczą pojawiające się stopniowo szczegółowe objaśnienia.



Ryc. 9. Końcowa faza wizualizacji algorytmu Euklidesa

### *Czynności ucznia*

Pracę z programem Algorytm Euklidesa poprzedza się przypomnieniem metody znajdowania największego wspólnego dzielnika dwóch liczb. Po uruchomieniu ekranowej projekcji programu uczniowie próbują odpowiadać na pytania dotyczące realizowanych w programie operacji. Oto przykłady formułowania sekwencji pytań problemowych: Co to znaczy Pobierz X1 oraz Pobierz X2? Co oznacza  $a := X1$  oraz  $b := X2$ ? Do czego niezbędne jest sprawdzenie warunku  $a > b$ ? Co trzeba wykonać, aby uporządkować liczby  $a$  i  $b$ , jeśli warunek  $a > b$  nie jest spełniony? Jakie działanie ma być wykonane, jeśli już spełniony jest warunek  $a > b$ ? Co trzeba wykonać, jeśli reszta jest różna od zera? Kiedy reszta z dzielenia jest zerowa, która ze zmiennych wyznacza NWD? Które zmienne trzeba wykorzystać we wzorze obliczającym NWD? Jakie znaczenie mają dwa ostatnie kroki w grafie tegoż algorytmu?

### *Komentarz nauczyciela*

Szczególne uwagi należy zwrócić na informatyczne aspekty algorytmizacji, takie jak: rozłożenie działań na elementarne kroki, wykorzystanie funkcji wprowadzania danych i wyświetlania wyników, przewidywanie ewentualnego błędu wprowadzenia najpierw mniejszej z liczb, użycie klasycznej metody wymiany wartości pomiędzy dwiema zmiennymi za pomocą zmiennej pomocniczej, skończone zapętlenie algorytmu z funkcją warunkową, efektywne wykorzystanie fragmentu pętli do dwóch różnych zadań oraz skorzystanie z dostępnej funkcji modulo, obliczającej tylko resztę z dzielenia.

## IMPLEMENTACJA POMYSŁU

Implementacja w dziedzinie informatyki ma dwa znaczenia. W głównym ujęciu jest to proces wykonania w postaci fizycznej czegoś, co przedtem było tylko w postaci teoretycznej. Najczęściej oznacza praktyczną realizację koncepcji programistycznej, co polega nie tylko na przejściu od algorytmów do procedur, lecz także na całościowym rozwiązaniu wszystkich zagadnień związanych z tworzeniem aplikacji, takich jak: *interfejs użytkownika, funkcje menu, sterowanie obiektowo-zdarzeniowe, taktowanie procedur zmienną częstotliwością zegara, wizualizację graficzną i animację, symulację procesów i monitorowanie parametrów, strukturyzację i archiwizację danych*. Wynika stąd, że implementacja pomysłu może być zwieńczeniem zaawansowanego procesu kształcenia informatycznego na poziomie licealnym. Wtórne znaczenie słowa „implementacja” używa się do nazywania programu, który powstał właśnie jako wdrożenie konkretnej koncepcji.

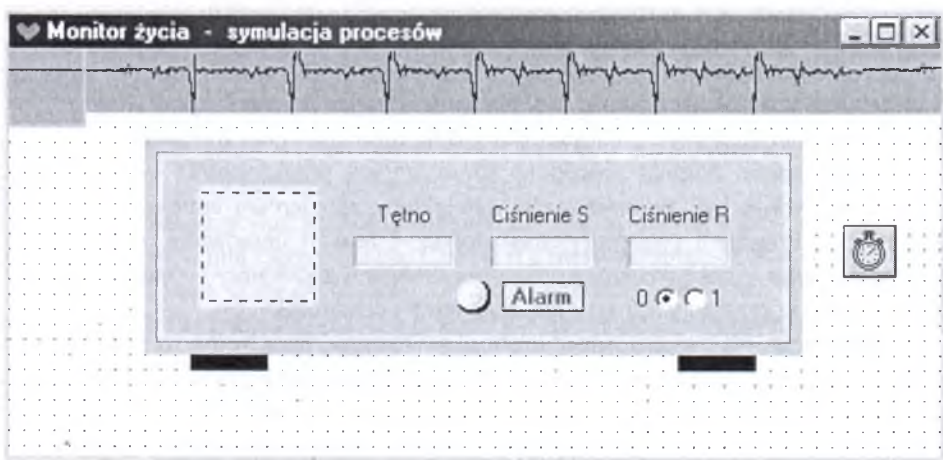
### Przykład 10. Monitor życia

Zadaniem ucznia jest wykonanie programu symulującego przebieg głównych procesów życiowych (tętno i ciśnienie) i pracę urządzenia kontrolującego stan pacjenta z funkcją alarmowania przy przekroczeniu wartości krytycznych. Do realizacji tak zaawansowanego zadania konieczne jest użycie narzędzia programistycznego. Najlepszym do tego celu jest MS Visual Basic, przy czym zupełnie wystarcza starsza wersja edukacyjna 3.0. Zajęcia zalecane są do przeprowadzenia w klasie licealnej o profilu informatycznym.

#### Czynności ucznia

Na wcześniejszych zajęciach z grafiki komputerowej uczniowie powinni przygotować obrazki na których przedstawione zostały kolejne cykle elektrokardiogramu (EKG). Nauczyciel nie powinien wydawać szczegółowych instrukcji, w jaki sposób uczeń ma coś wykonywać, lecz może jedynie inspirować i organizować pracę uczniów, definiując kolejne fazy postępowania programistycznego i stawiając uczniom pytania naprowadzające na rozwiązanie problemu. Oto szkic takiego trybu zajęć:

- Rozpoczynasz programowanie obiektowo-zdarzeniowe – *Co najpierw zaprojektujesz, obiekty czy zdarzenia?* – (obiekty).
- Obiekt imitujący przyrząd pomiarowy ma wyświetlać obraz oscylograficzny i trzy parametry liczbowe – *Jak mógłby wyglądać taki przyrząd?* – (szkic).
- Ustalmy, co powinno znaleźć się w oknie aplikacji – *Z jakich elementów powinien być zbudowany przyrząd?* – (obudowa, pole oscylogramu, trzy pola informacyjne z etykietami, lampka z napisem „Alarm”, włącznik / wyłącznik).



Ryc. 10. Przykładowy projekt okna z formantami imitującymi Monitor życia

- Uzgodnijmy nazewnictwo obiektów – *Jakie proponujecie nazwy?* – (np.: Okno, Panel, Obraz, Obraz1, Obraz2..., Napis1, Napis2..., NapisAlarm, Lampka, Tetno, CisnienieS, CisnienieR, Zegar, Zasilanie0, Zasilanie1 itd.).

**Konstrukcja okna.** Zbudujcie wszystkie obiekty z tymi ustalonymi nazwami – *Jakie typy obiektów zastosujecie?* – (pola obrazowe typu Picture oraz Image, pola tekstowe z etykietami, zegar, przyciski opcji, ramki, linie itp.).

- Przyjmijmy nazwy pewnych wartości stałych, definiujących kolory zapalone/wygaszone/alarmujące – *Jak nazwiemy zmienne charakteryzujące parametry mierzone u pacjenta i dodatkowe zmienne definiujące tendencje?*

- Wpiszmy obsługę ustawiania początkowych wartości podczas uruchamiania programu – *Jakich funkcji użyć trzeba do wygenerowania liczb losowych?* – (nauczyciel podpowiada, w jakim przedziale mają być losowane liczby).

- Przystępujemy do definiowania procedur – *Jak nazwalibyście procesy, jakie zachodzą na oddziale intensywnej opieki medycznej?* – (czuwanie, mierzenie, sygnalizacja, alarmowanie, monitorowanie, wyświetlanie na ekranie itp.).

- W pracowni komputerowej nie mamy pacjenta ani czujników pomiarowych – *Jak nazywa się sytuację, kiedy komputer imituje procesy, zastępując różne inne urządzenia?* – (symulacja).

- Uzgodnijmy nazewnictwo procedur na podstawie wcześniej wymienionego słownictwa (symulator, monitor, sygnalizator) oraz utwórzmy pętlę czasową – *Jaką procedurę wpisać do zegara, ażeby rytm pomiarowy był związany z tętnem pacjenta a 8-krotnie szybciej przebiegała animacja obrazu EKG?* – (funkcja interwałowa zegara z wydzieleniem ośmiu faz).

- Wpiszmy do pierwszej z faz wywołanie trzech głównych procedur.

**Symulacja.** Napisz procedurę symulacji procesów z losową tendencją zmian tętna i ciśnienia, z większym prawdopodobieństwem spadku tych wartości – *Jaki wzór zastosować, aby przeliczyć częstotliwość tętna (liczbę uderzeń serca na minutę) na interwał czasowy (okres pulsu wyrażony w milisekundach)?*

- Napisz procedurę wyświetlania aktualnych wartości tętna i ciśnienia w polach tekstowych – *Zastanów się, czy polom tekstowym można bezpośrednio przypisywać wartości liczbowe?* – (Visual Basic dokonuje konwersji formatu).

**Sygnalizacja.** Napisz procedurę sprawdzającą przekroczenie wartości tętna i ciśnienia. Ma to być sygnalizowane dźwiękiem, zapaleniem lampki oraz zmianą koloru pól. Fragment procedury musi dotyczyć reakcji urządzenia na śmierć kliniczną pacjenta – *Jaką strukturę posiadają funkcje warunkowe?* – (nauczyciel podpowiada, jakie przyjąć wartości graniczne tętna i ciśnienia).

**Aktywacja.** Napisz procedurę imitującą włączenie urządzenia za pomocą przycisku, z rozjaśnieniem barwy tła pól tekstowych i pola oscylografu – *Jak uruchomić zegar wywołujący procedury cykliczne o określonym interwale?*

- Napisz procedurę imitującą wyłączenie urządzenia z zatrzymaniem zegara, z wygaszeniem obrazów i wyczyszczeniem pól tekstowych – *Jak wyczyścić pole obrazowe?* – (np. przepisać zawartość z pola, które nie zawiera obrazu).

**Animacja.** Uzupełnij procedurę zegara o sekwencje animacji, której wynikiem będzie poziome przewijanie wykresu elektrokardiogramu w polu obrazowym. Dodaj migotanie lampki alarmu w rytmie tętna.

- Uruchom program, sprawdź prawidłowość i obserwuj efekty działania.
- Zapisz projekt oraz utwórz plik wykonywalny o nazwie: Monitor.exe

### *Zadanie domowe*

Wprowadź modyfikację animacji w taki sposób, aby amplituda wykresu elektrokardiograficznego malała proporcjonalnie wraz ze spadkiem ciśnienia.

### *Komentarz nauczyciela*

Nauczyciel zwraca szczególną uwagę na kompleksowość implementacji, ponieważ współczesne programowanie nie jest wyłącznie pisaniem procedur. W pierwszej fazie projektowania obiektów dopuszczalne są odmienne koncepcje graficzne, lecz z zachowaniem zgodnego nazewnictwa formantów, aby później przy tworzeniu procedur nie było zamieszania utrudniającego prowadzenie zajęć. Z tego samego względu w fazie definiowania zmiennych oraz w fazie deklarowania procedur warto także ustalić jednolite nazewnictwo, przy czym propozycje nazw powinny być ustalane metodą „burzy mózgów”.

Wypełnianie procedur powinno być w miarę możliwości wykonywane samodzielnie przez ucznia – w postaci małych porcji kodu, z natychmiastowym korygowaniem ewentualnych błędów składniowych języka, sygnalizowanych podczas przejścia do edycji następnego wiersza. Jednakże ogólny zarys projektu w aspekcie tego co ma być wyświetlane (wartość tętna i ciśnienia), co animowane (obraz EKG), co i czym sygnalizowane (przekroczenie wartości granicznych), a zwłaszcza, co ma być symulowane – musi być uczniom jasno przedstawiony. Należy natomiast dopuścić różnorodne, własne koncepcje uczniowskich procedur i funkcji, jeśli tylko spełniają one kryteria poprawności i efektywności. Dotyczy to przede wszystkim procedury symulacji, która jest najtrudniejsza do napisania, lecz która ma imitować procesy stochastycznie niestacjonarne, a tym samym – w różnorodności jej rozwiązań tkwi zgodność z realiami procesów życiowych.

Warto zaakcentować komputerową możliwość działania współbieżnego – symulacji, monitowania i sygnalizacji. Chodzi o generowanie przez procedury zarówno sygnałów, które w rzeczywistym pomiarze pochodzą z zewnątrz, tj. z czujników podłączonych do pacjenta, jak też o podział i przydzielanie czasu na wykonanie procedur animacji oraz testowanie wartości krytycznych. W trakcie projektowania obiektów, w których później odbywać się będzie animacja, należy koniecznie zasugerować uczniom użycie formantu typu Image (o nazwie Obraz) umieszczonego wewnątrz formantu typu Picture (o nazwie Pole). Ułatwi to rozwiązanie zadania domowego.

## ZAKOŃCZENIE

Informatyka wykształciła już tak duży zasób specjalistycznych pojęć oraz metod i technik budowania systemów, że opanowanie języka tej dziedziny wiedzy oraz nabycie umiejętności posługiwania się narzędziami i technologiami staje się kluczem dającym wstęp do elitarnej klasy zwanej *społeczeństwem informacyjnym kognitariuszy*. Do pierwotnych zadań informatyki doszły nowe, znacznie trudniejsze. Nie wystarcza już przetwarzanie informacji, niezbędne jest szersze budowanie systemów wiedzy zarówno w formach reprezentacji ikonicznych, jak i symbolicznych, a także zaawansowanych systemów sztucznej inteligencji: potrafiących interpretować fakty, analizować wydarzenia, współtworzyć kierunki rozwoju, kreować przemiany społeczne, wspierać procesy decyzyjne itd.

Informatyczne reprezentacje świata oraz interpretacje otaczającej nas rzeczywistości niosą z sobą znaczną, interdyscyplinarną wartość poznawczą. Pojęcia pierwotnie ukształtowane przez informatyków stają się ważne także w innych obszarach rozwoju osobistego i społecznego. Umiejętność selekcji, porządkowania, rangowania, szukania relacji, kreowania innowacji, obrazowania zjawisk, prezentacji, tworzenia procedur itp. powinna dotyczyć przecież postępowania w każdej sferze ludzkiej działalności. Aby jednak właśnie w sensowny sposób postępować, trzeba zrozumieć istotę owej swoistej logiki własnych działań. Każda czynność w umyśle ludzkim przekodowywana jest na jej werbalną reprezentację, stąd dla postępowania efektywnego niezbędne jest poznawanie poprzez bezpośrednie działanie połączone z desygnowaniem nazw operacyjnych. W ten sposób zamyka się koło powiązań pomiędzy definicjami czynnościowymi a definicjami pojęciowymi. Mechanizm ten umożliwi kształtowanie, zwłaszcza w młodym wieku, nawyków do działania racjonalnego.

W niniejszym opracowaniu zostały przedstawione przykłady kształtowania poprzez działanie zaledwie kilku głównych pojęć informatycznych i niejako w tle pojęć towarzyszących, w liczbie zależnej od słownictwa używanego na lekcji przez nauczyciela. Warto jednak na podstawie tego wzorca skonstruować autorski program nauczania informatyki, wprowadzając według tych samych zasad wszystkie dalsze pojęcia, takie jak: *edycja* (np. dźwięku, obrazu, filmu), *eksploracja zasobów*, *iteracja*, *rekurencja*, *sortowanie*, *programowanie*, *kompilacja*, *translacja*, *kodowanie*, *redundancja*, *logowanie*, *transmisja*, *interpretacja*, *predykcja błędów*, *interakcja*, *instalacja*, *animacja*, *ekspozycja*, *kontekstualizacja hipertekstu* itd. Wśród tych przykładowych pojęć znajdują się zarówno zagadnienia łatwe, jak i trudniejsze. Oczywiście, należy zawsze brać pod uwagę poziom, zapisy zadań, celów i treści określone w podstawach programowych nauczania ogólnego bądź profilowanego, a ponadto rzeczywiste możliwości percepcyjne własnych uczniów.