

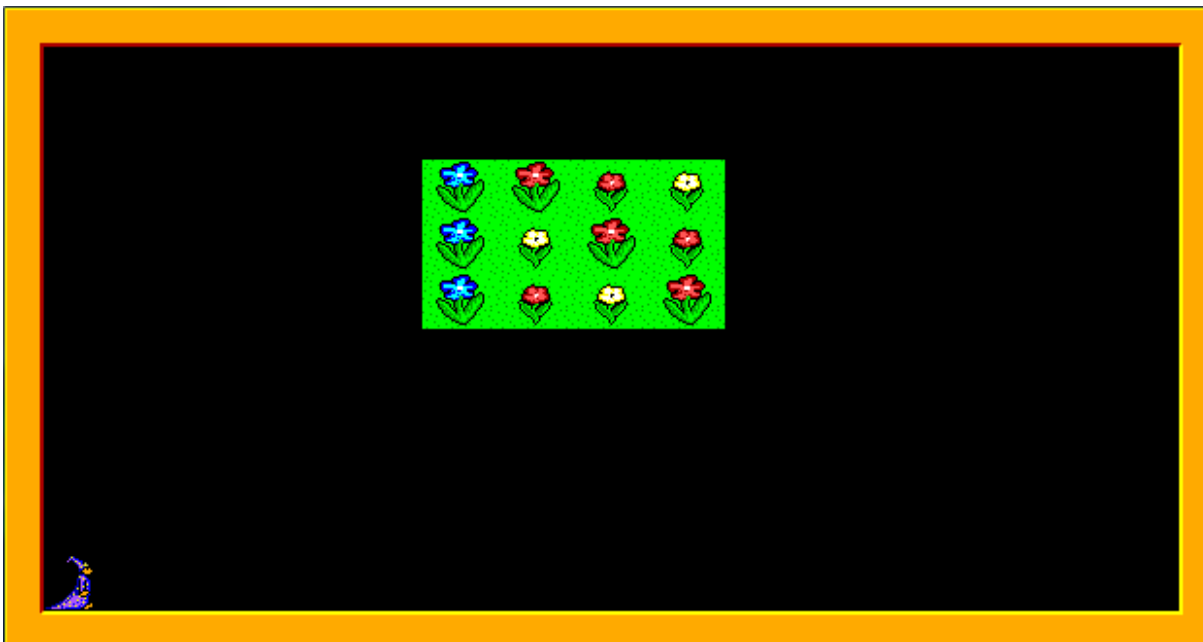
Programowanie w Balcie klasa VII

Zadania z podręcznika strona 127 i 128

Zadanie 1/127



Efekt wywołania programu:



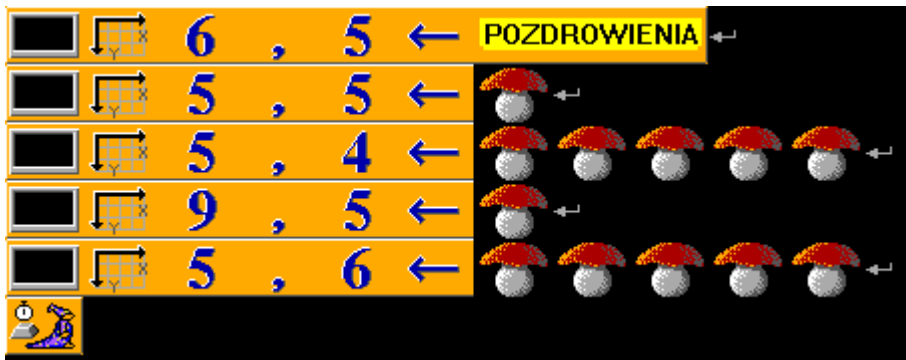
Zadanie 2/127



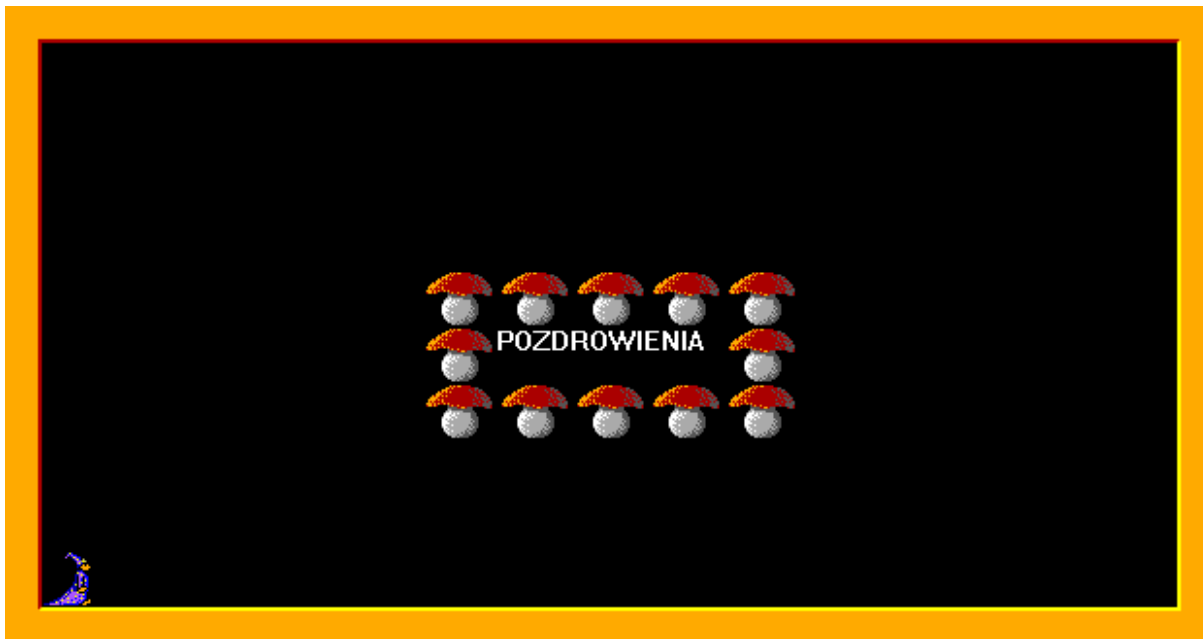
Efekt wywołania programu:



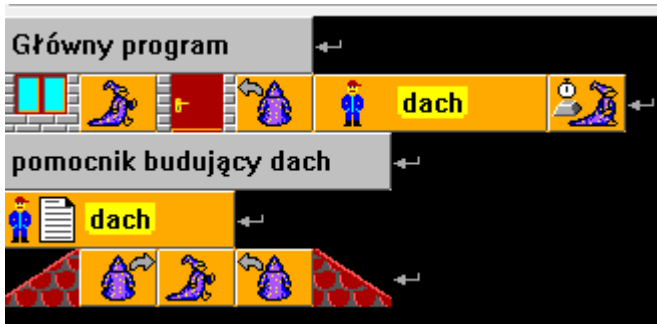
Zadanie 3/127



Efekt wywołania programu:



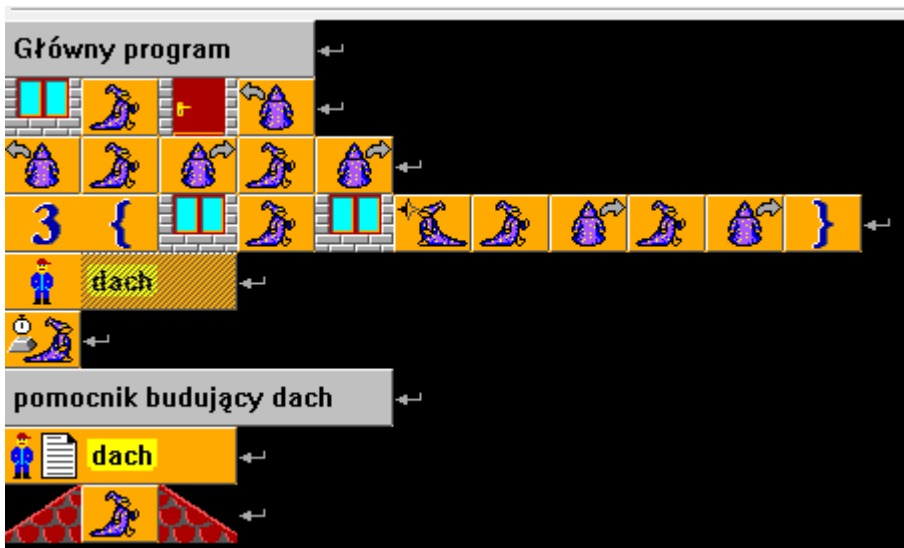
Zadanie 4/127



Efekt wywołania programu:



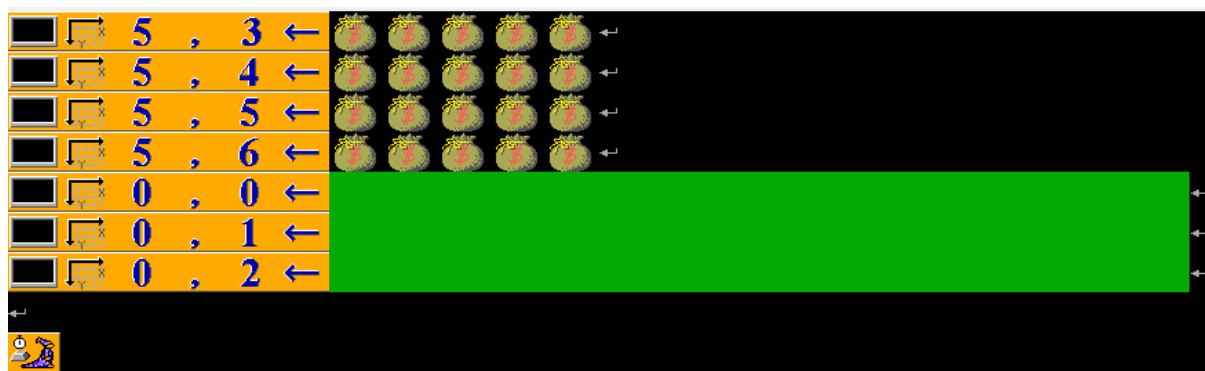
Zadanie 5/127



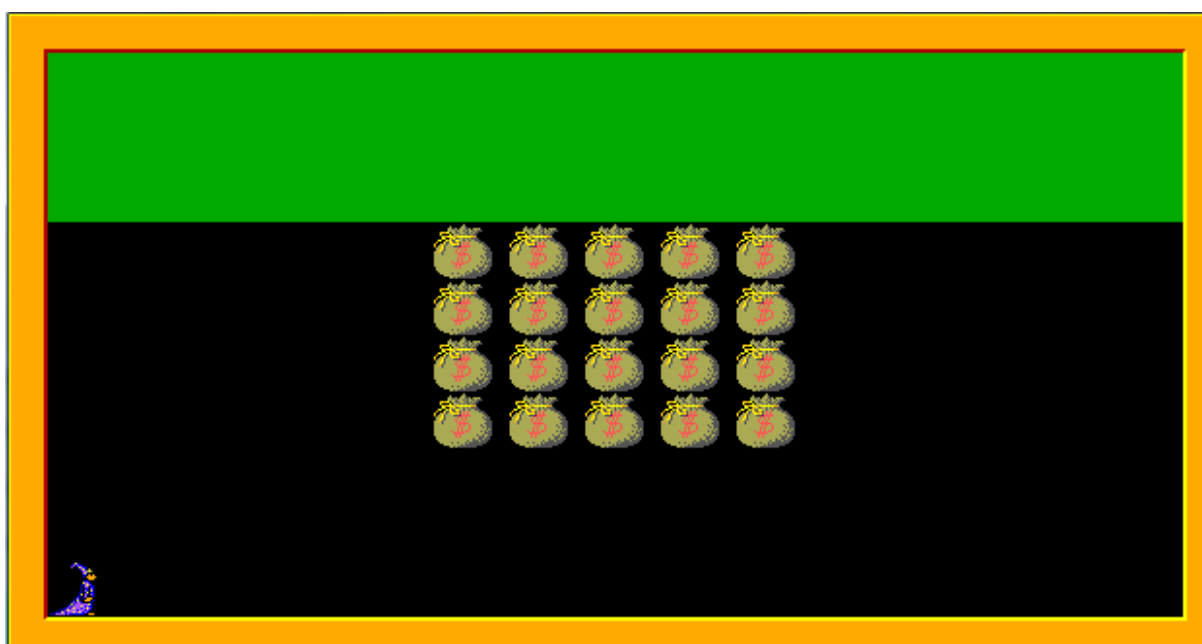
Efekt wywołania programu:



Zadanie 6/127



Efekt wywołania programu:



Ten sposób pisania programu nie ma sensu!!!. Choć oczywiście może być poprawny. To jest tylko fragment ale i tak jest długi. A będzie jeszcze dużo dłuższy. Spróbujmy inaczej.

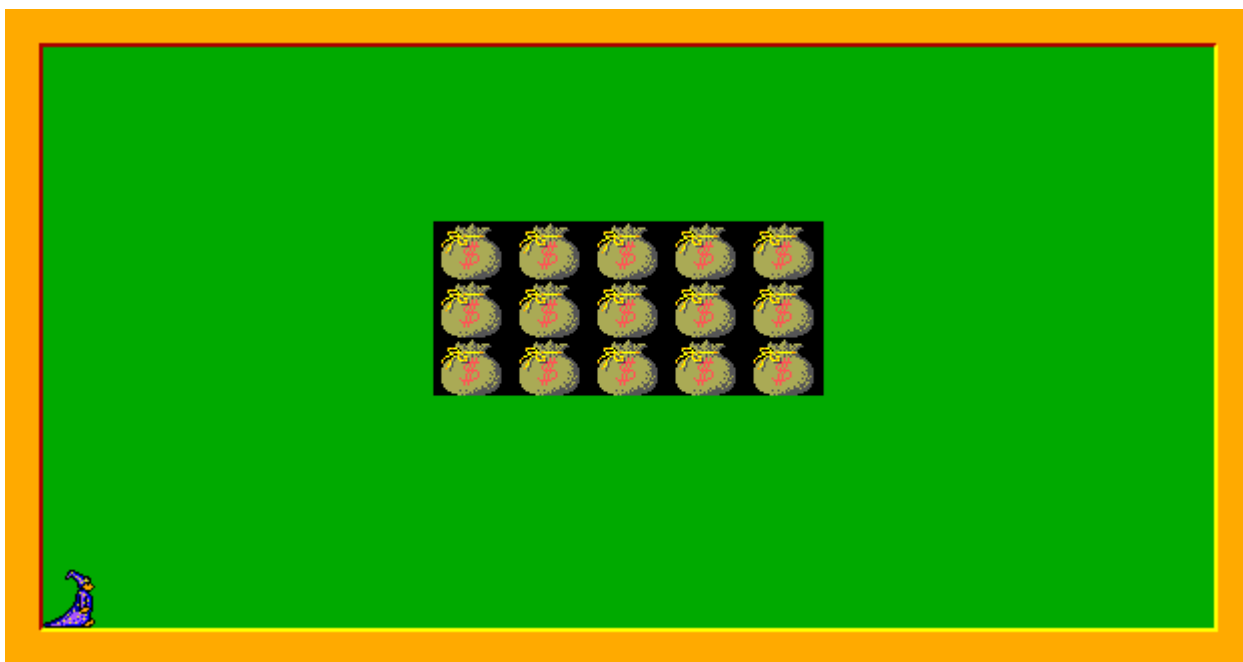
```

4 { [ ] 5 , aA ← [ ] [ ] [ ] [ ] [ ] aA ← aA + 1 } ←
3 { [ ] 0 , bB ← [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] bB ← bB + 1 } ←
4 { [ ] 0 , cC ← [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] cC ← cC + 1 } ←
4 { [ ] 1 0 , dD ← [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] dD ← dD + 1 } ←
eE ← 6 ←
4 { [ ] 0 , eE ← [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] eE ← eE + 1 } ←
0 ←

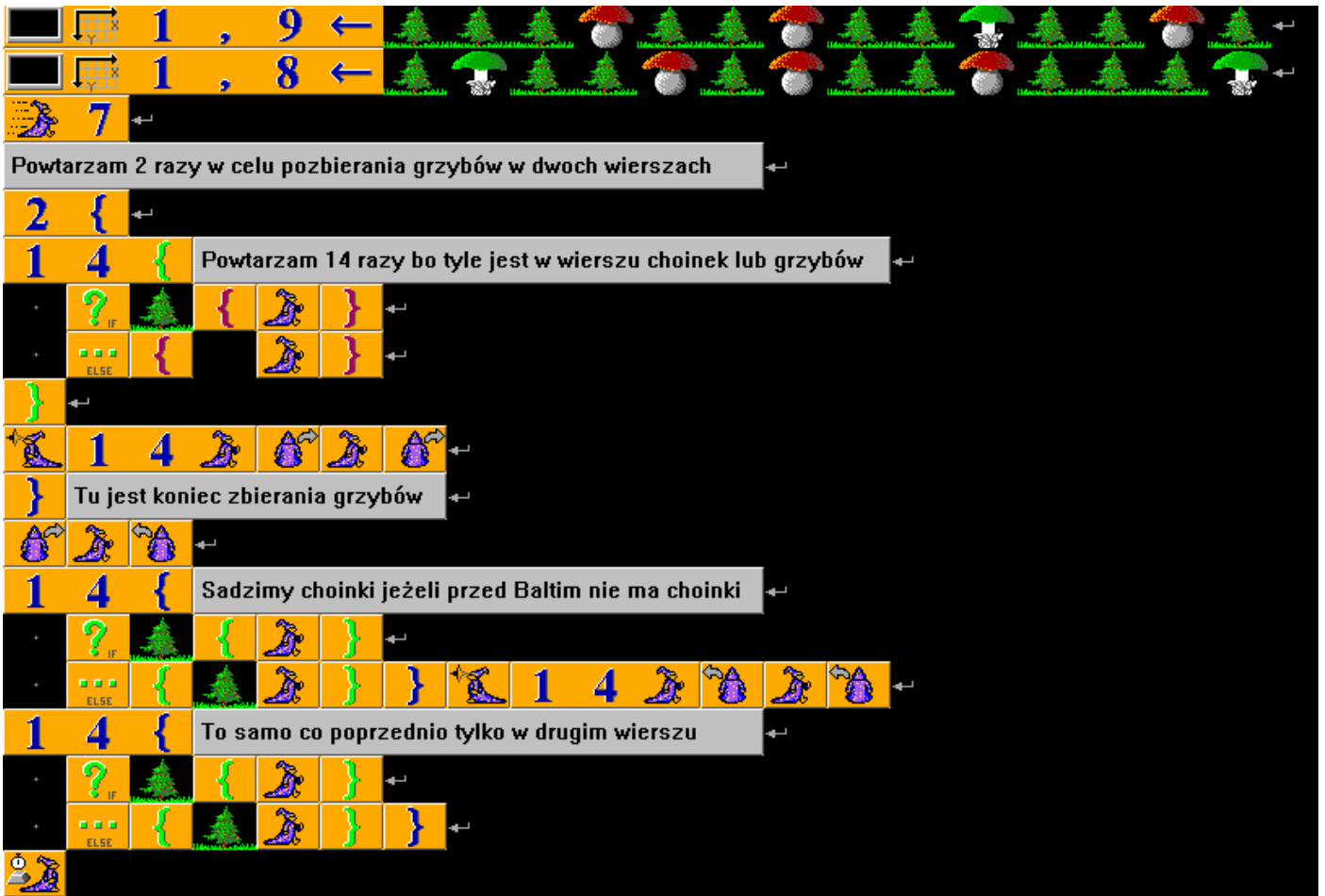
```

Zmienna a ma początkową wartość 3 (bo boisko będzie w 3, 4, 5, 6 wierszu). Zwiększa się za każdym powtórzeniem o 1 by boisko było tworzone w kolejnym wierszu. Przypominam wiersze są ponumerowane od góry: 0, 1, 29. Podobnie jest z wykorzystaniem pozostałych zmiennych.

Efekt wywołania programu:



Zadanie 8/128



Efekt wywołania programu:

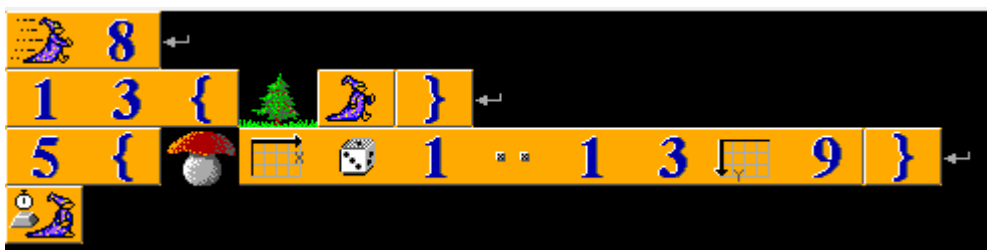
Pierwsza część



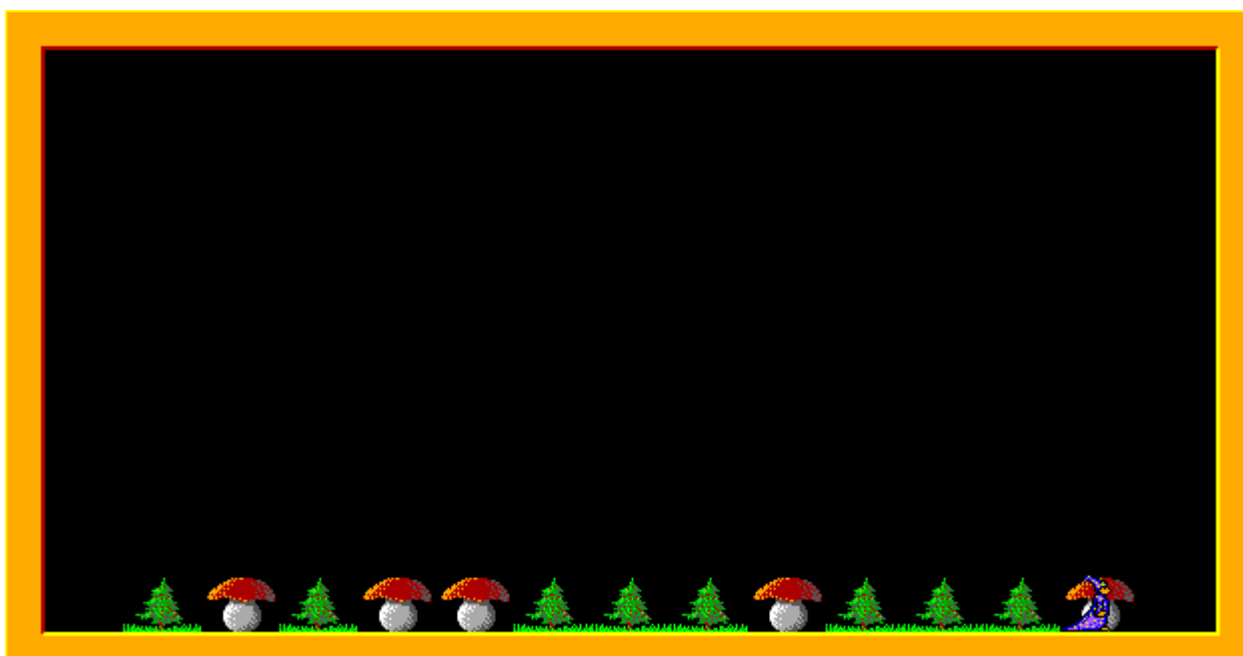
Druga część



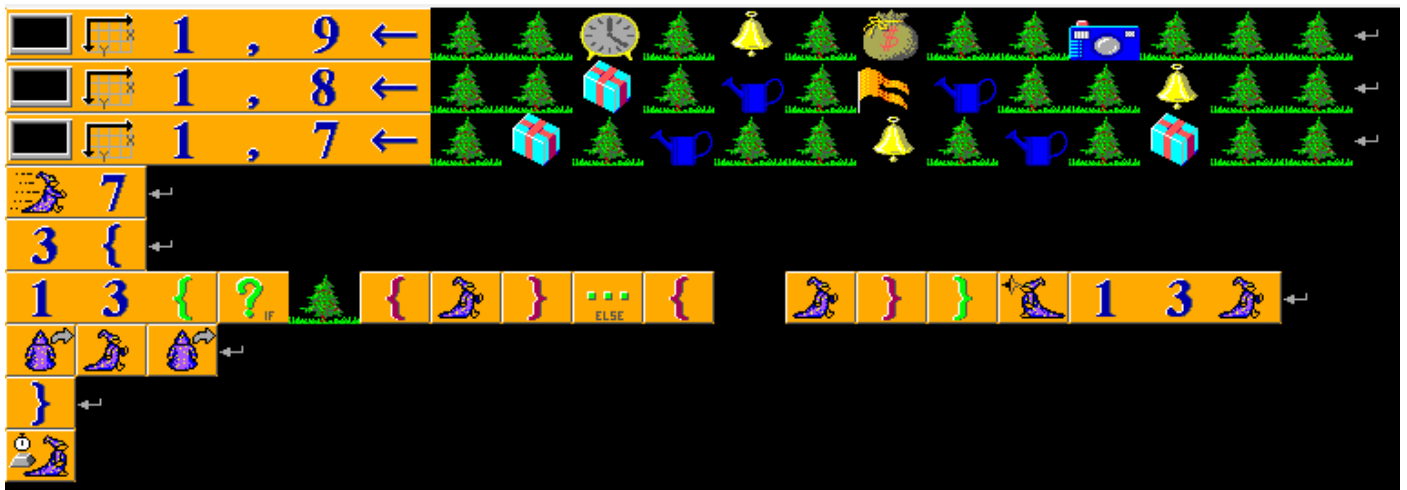
Zadanie 9/128



Efekt wywołania programu:



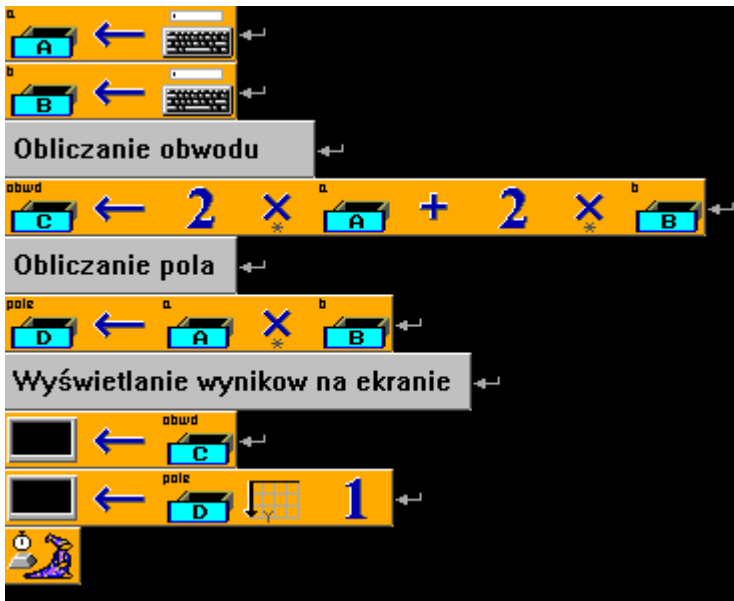
Zadanie 10/128



Efekt wywołania programu:



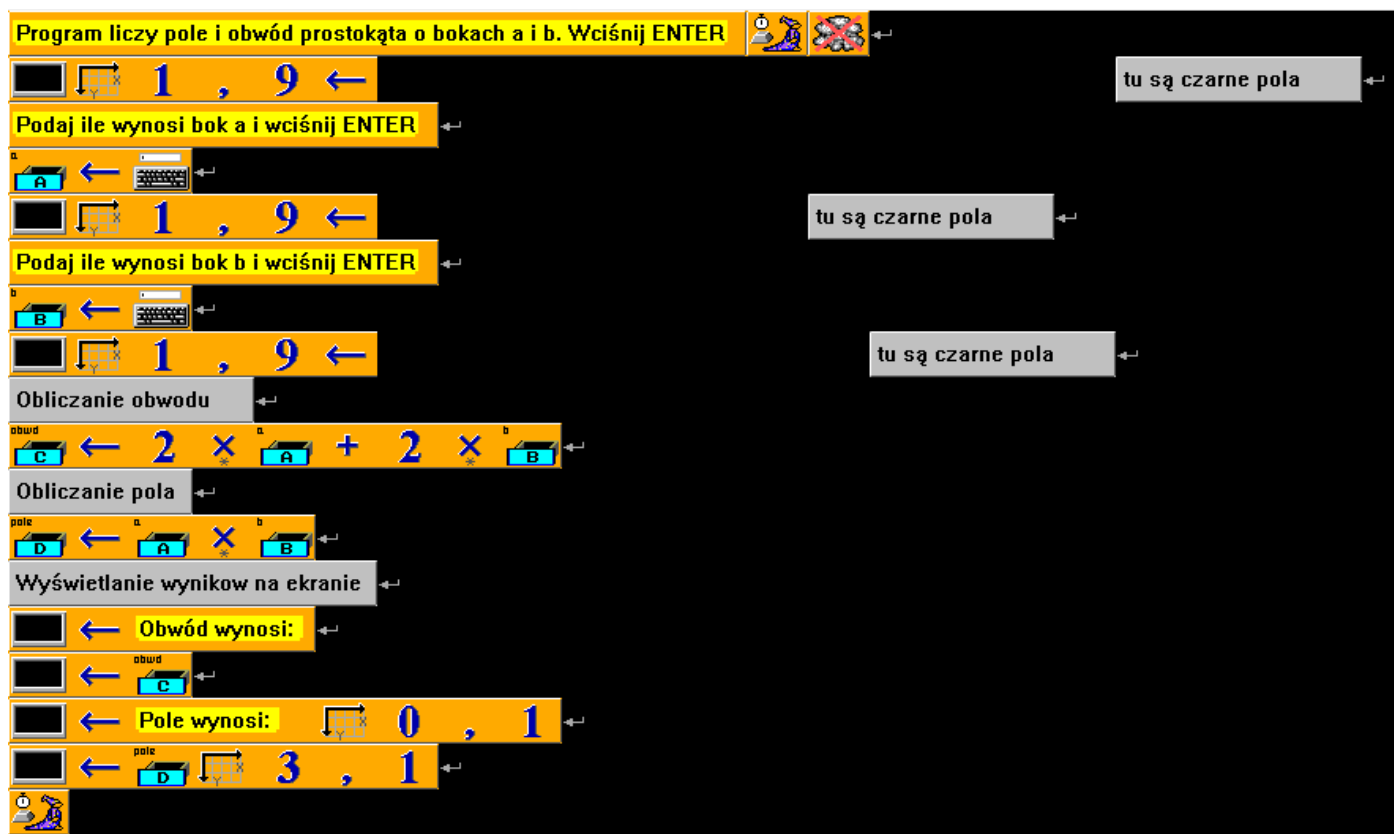
Zadanie 11/128



Efekt wywołania programu dla 2 i 3:



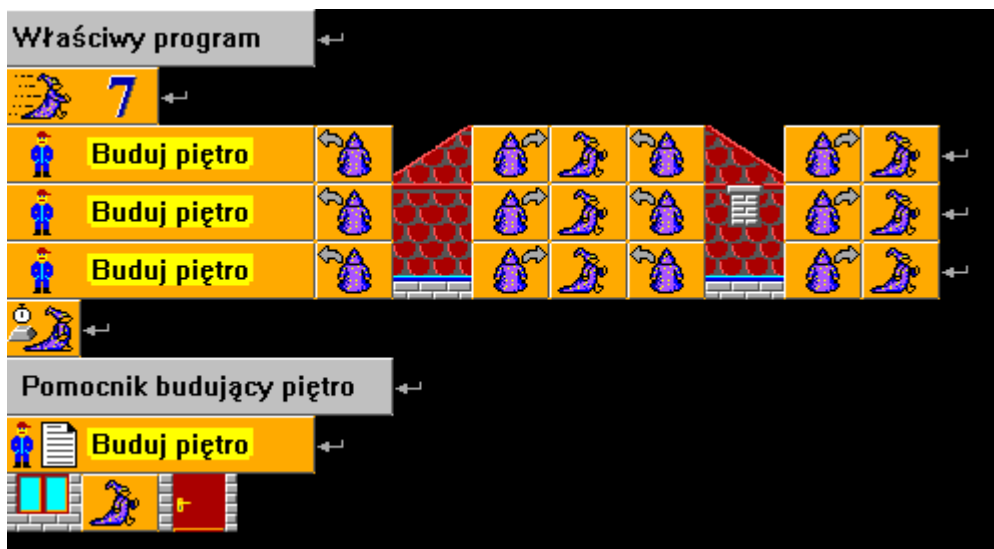
Program można usprawnić, tak by był czytelniejszy.



Czarne kwadraty mają ukryć poprzedni tekst.

Efekt:





Efekt wywołania programu:



Zadanie 15/128 Suma n-kolejnych liczb naturalnych

Podaj ile liczb mam sumować:

Zmienna ileliczb oznacza ile kolejnych liczb naturalnych ma być sumowanych

Zmienna suma ma początkową wartość 0

Została tu wprowadzona pętla WHILE, działa ona dopóki warunek jest spełniony

w naszym przypadku dopóki $a > 0$

WHILE $ileliczb > 0$

$suma \leftarrow suma + ileliczb$ sumę zwiększamy o liczbę podaną przez użytkownika

$ileliczb \leftarrow ileliczb - 1$ liczbę użytkownika zmniejszamy o 1

Suma liczb wynosi: 1

5, 1

Efekt wywołania programu dla 6 kolejnych liczb. Kolejno są dodawane: $6 + 5 + 4 + 3 + 2 + 1$

Suma liczb wynosi: 21

Podaj ile liczb mam sumować:

Zadanie 16/128

7

1 0 0 {

1 0 0 {

1 0 0 {

1 0 0 {

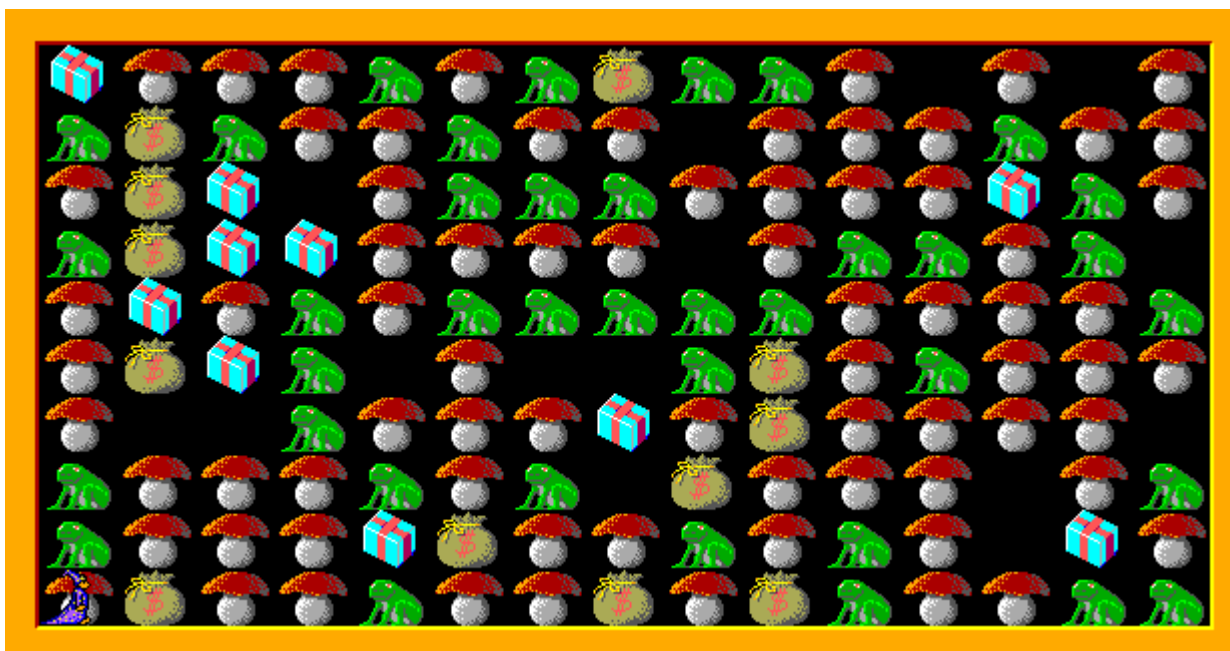
1 5 1 0 }

1 5 1 0 }

1 5 1 0 }

1 5 1 0 }

Efekt wywołania programu:



Inne ciekawe zadania matematyczne.

Zadanie 1

Program wyświetla informację, czy podana liczba jest parzysta, czy nieparzysta.

Program sprawdza czy podana liczba jest parzysta czy nieparzysta. Podaj liczbę.

Rozwiązaniem jest sprawdzanie czy reszta z podziału liczby przez 2 jest równa 0 czy nie

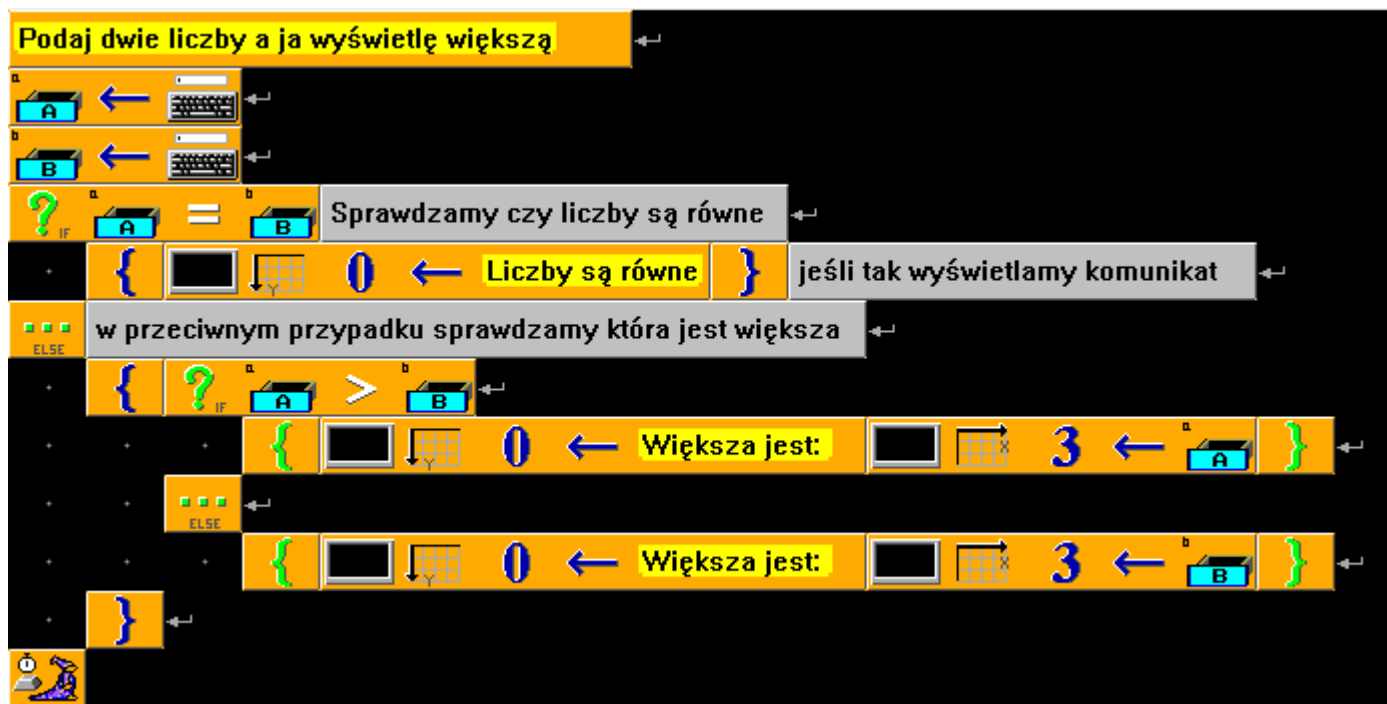
% ten operator daje w wyniku resztę z dzielenia.

Liczba jest parzysta jeżeli reszta z dzielenia = 0 to jest to liczba parzysta

Liczba jest nieparzysta w przeciwnym wypadku jest to liczba nieparzysta

Zadanie 2

Program podaje, która z dwóch liczb jest większa lub informację liczby są równe.

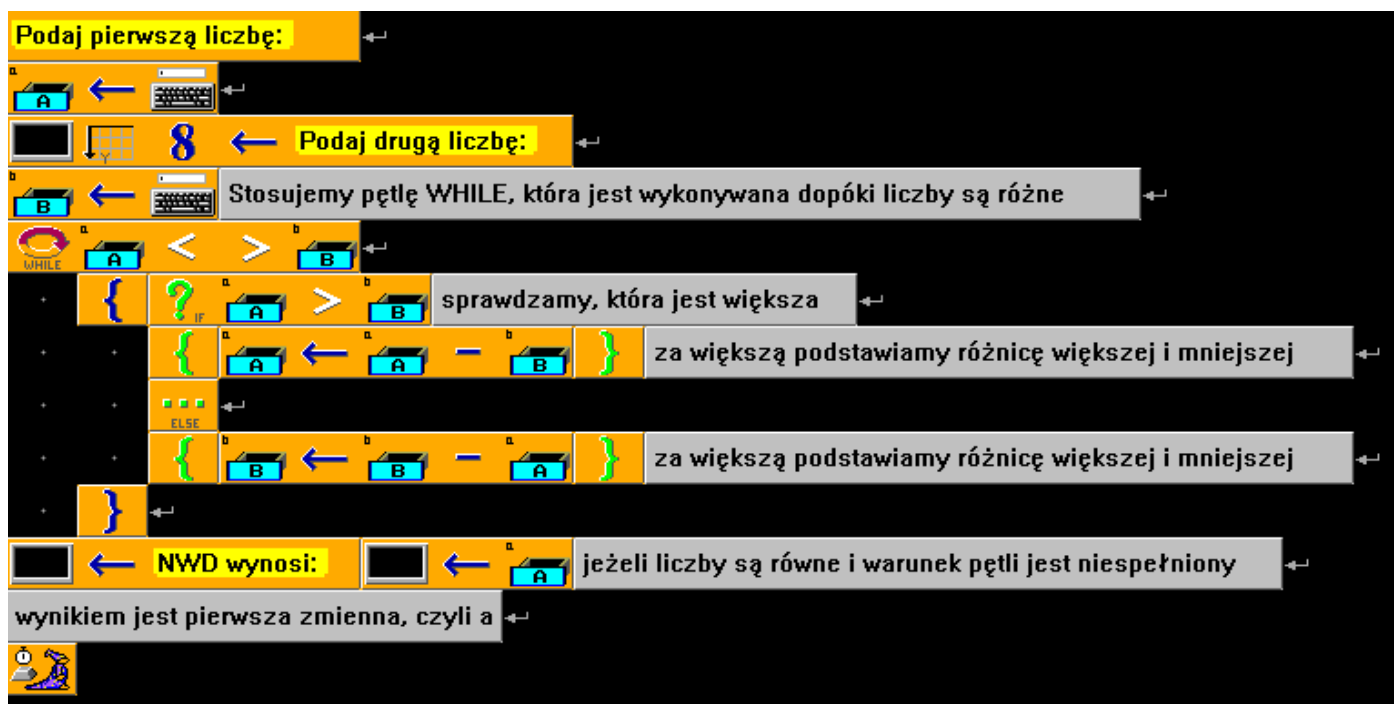


Dla czytelności kodu warto stosować wcięcia – czarny kwadrat z kropką. Wcięcia stosujemy przy pętlach i warunkach.

Zadanie 3.

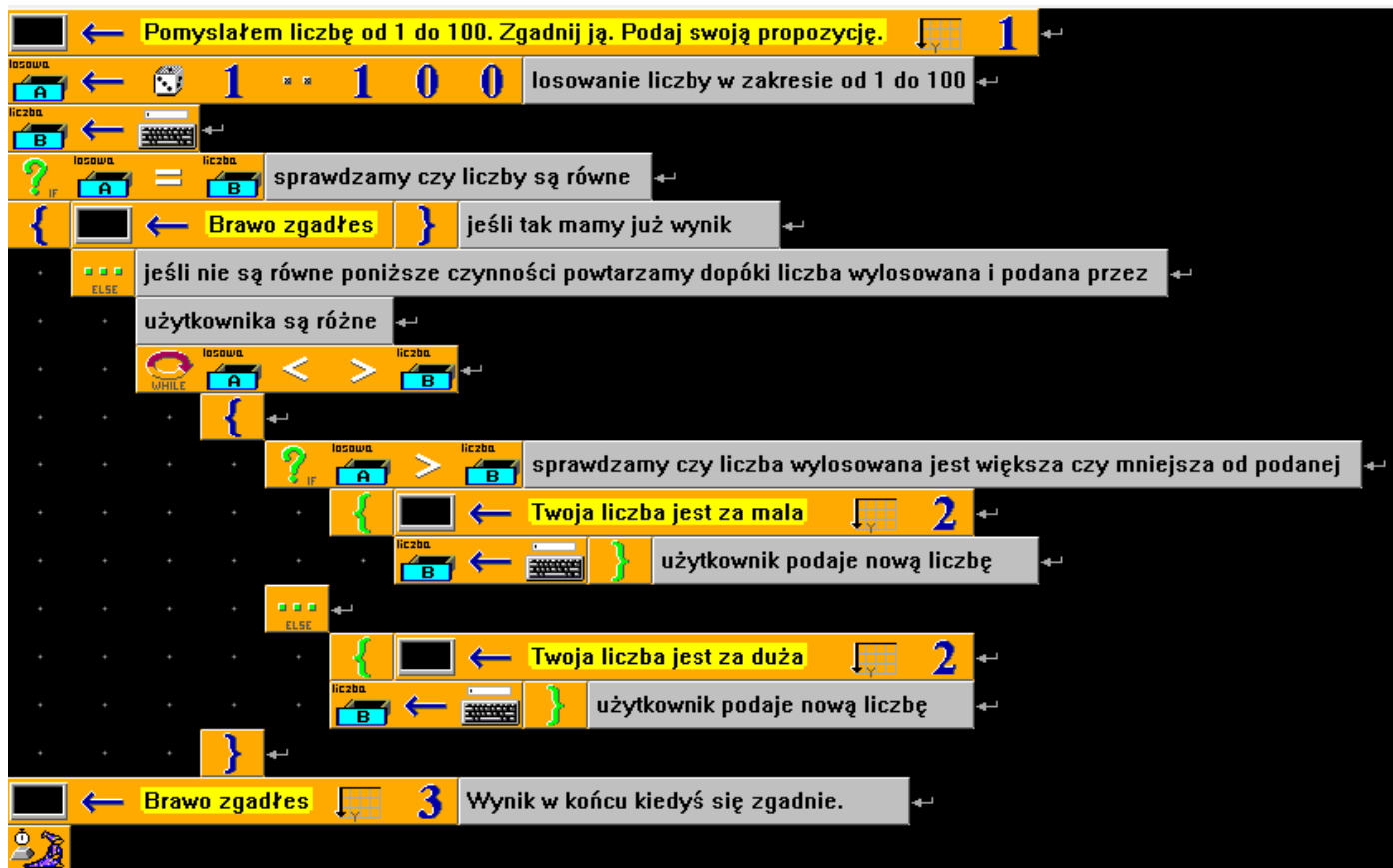
Obliczanie największego wspólnego dzielnika dwóch liczb. (NWD). Korzystamy z algorytmu Euklidesa.

Algorytm jest opisany w podręczniku strona 116. Zadanie 10 dla zainteresowanych.



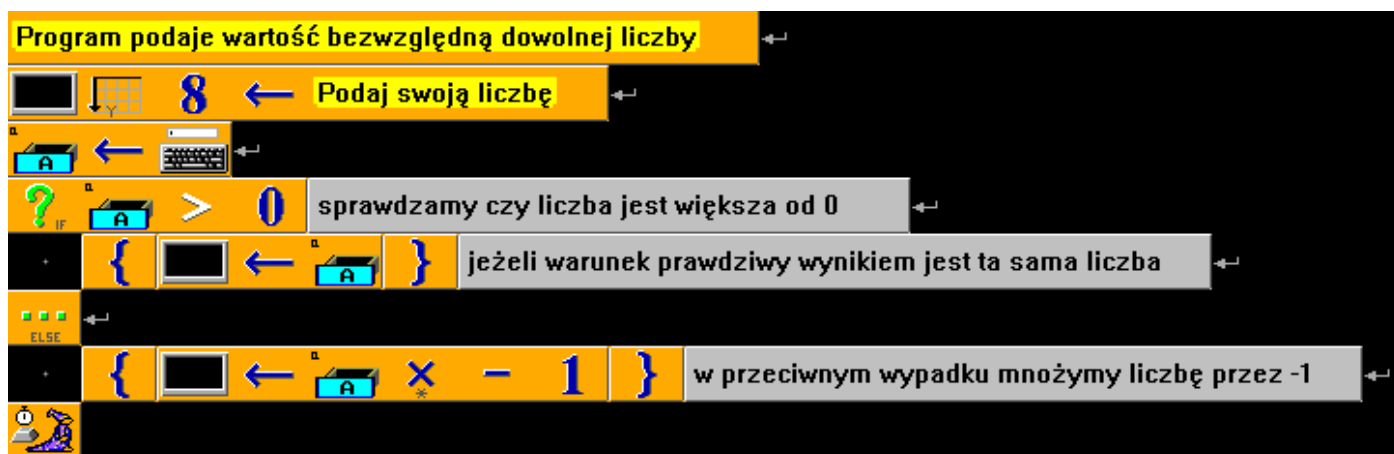
Zadanie 4.

Program losuje liczbę z zakresu od 1 do 100. Zadanie polega na odgadnięciu jej. Jeśli ją zgadniemy wyświetli się informacja „Brawo zgadłeś”. W przeciwnym wypadku wyświetli się informacja, że podana liczba jest za mała lub za duża. Program działa aż do odgadnięcia liczby.



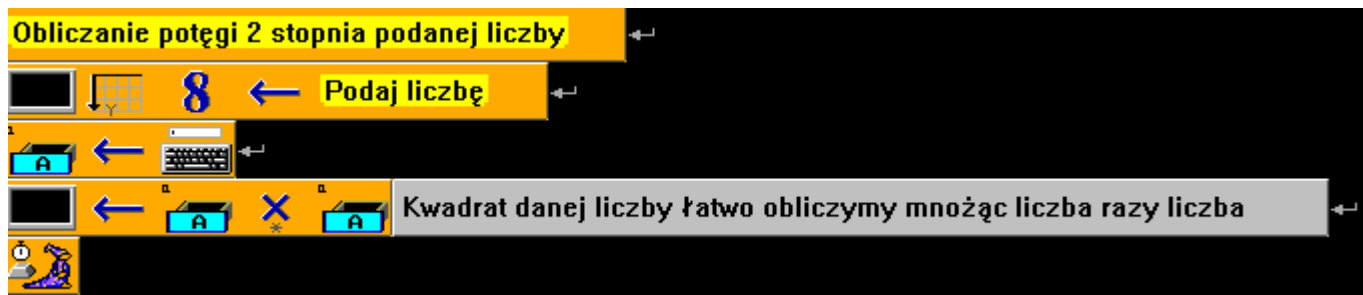
Zadanie 5

Program podaje wartość bezwzględną dla podanej liczby. Ważna informacja z matematyki – wartością bezwzględną liczby dodatniej jest ta sama liczba na przykład dla 6 jest to 6, wartością bezwzględną liczby ujemnej jest jej liczba przeciwna (o odwrotnym znaku) dla -6 jest to 6.



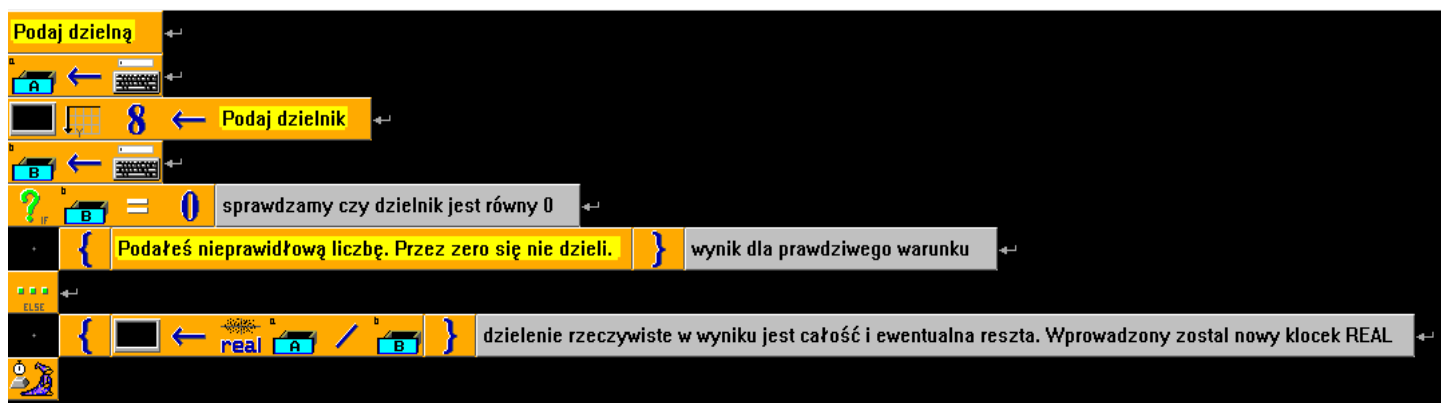
Zadanie 6.

Program podający potęgę drugiego stopnia podanej przez użytkownika liczby.



Zadanie 7.

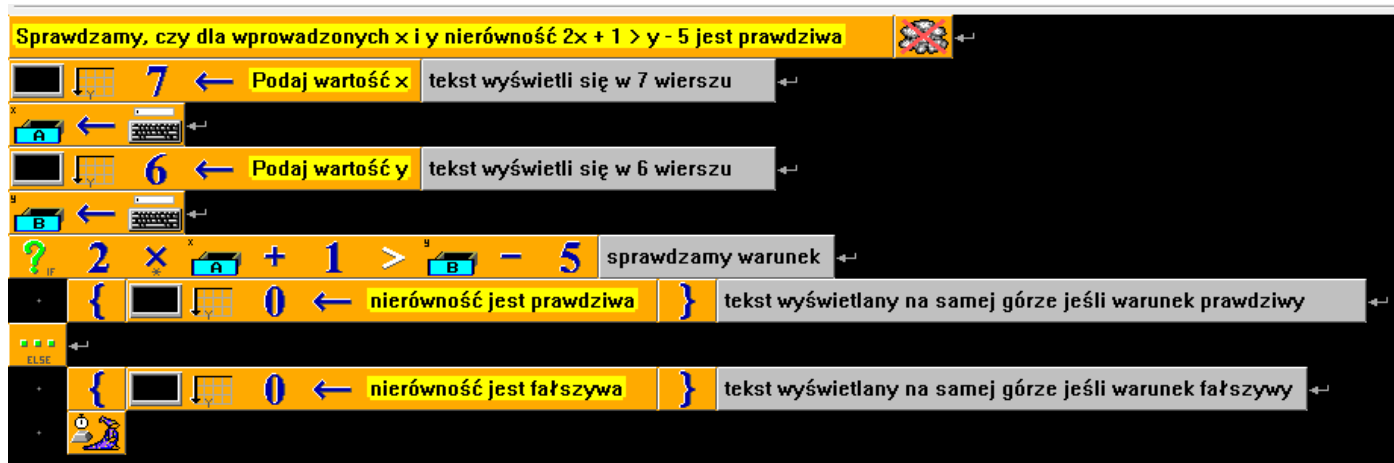
Program ma podać wynik (iloraz) dzielenia a przez b. Zadanie z pozoru łatwe, ale należy wziąć pod uwagę, że przez 0 się nie dzieli. Zatem b nie może być 0. Jeżeli użytkownik za dzielnik, czyli b poda 0 należy w wyniku wyświetlić informację „Podałeś nieprawidłowy dzielnik. Przez zero się nie dzieli”.



Zadanie 8.

Program, który sprawdza czy dla podanych przez użytkownika dowolnych x i y prawdziwa jest nierówność:

$2x+1 > y - 5$. Wynikiem jest informacja: nierówność jest prawdziwa lub nierówność jest fałszywa.



nierówność jest prawdziwa

Podaj wartość y

Podaj wartość x

Sprawdzamy, czy dla wprowadzonych x i y nierówność $2x + 1 > y - 5$ jest prawdziwa

Zadanie 9

Co ciekawe zwykłe dzielenie dowolnej liczby a przez b nie jest taką oczywistością jak mogłoby się wydawać. I pomijam tu warunek, że dzielnik czyli b musi być różny od zera, bo zakładam, że dzielnik, czyli b zgodnie z tym co nam mówi matematyka jest różne od zera żeby można było wykonać działanie. A jednak dzielenie a przez b może dać zaskakujące wyniki, często różne. Zobaczmy to na przykładzie. Jeżeli dzielna (a) wynosi 10 a dzielnik (b) 3 to wynikiem dzielenia $10/3$ może być: 3,3333.....3 lub 3 lub 1. I o co w tym chodzi? Wynik pierwszy to iloraz rzeczywisty (tu dodatkowo w okresie, czyli wynik jest liczbą nieskończoną), drugi wynik to ile jest całości (czyli trójek w 10 jest trzy), trzeci wynik to reszta z dzielenia (dlaczego 1, a nie 3? Bo $3*3$ całości = 9 a do 10 brakuje 1 a nie 3). Może trochę to skomplikowane ale to tylko matematyka na poziomie klas 1-3 i klasy 4. Zadaniem jest napisać program który dla a podzielić na b da trzy wyniki: wynik rzeczywisty, całkowity i resztę. Zakładam poprawność dzielnika, czyli b.

Różne wyniki dzielenia a/b, zakładam, że wprowadzone b jest poprawne

7 ← Podaj dzielną a

6 ← Podaj dzielnik b

5 ← Wynik rzeczywisty

5 , 5 ← real A / B należy dodać klocek REAL

4 ← Całość z dzielenia

5 , 4 ← A / B do dzielenia całkowitego używamy operatora / w języku C++ jest to //

3 ← Reszta z dzielenia

5 , 3 ← A % B zapis dzielenia z resztą to %, w C++ jest to ten sam operator

Efekt wywołania programu dla $a = 10$ $b = 3$:

```
Reszta z dzielenia      1
Całość z dzielenia     3
Wynik rzeczywisty      3,333333333333333
Podaj dzielnik b
Podaj dzielną a

Różne wyniki dzielenia a/b, zakładam, że wprowadzone b jest poprawne
```

Zadanie 10.

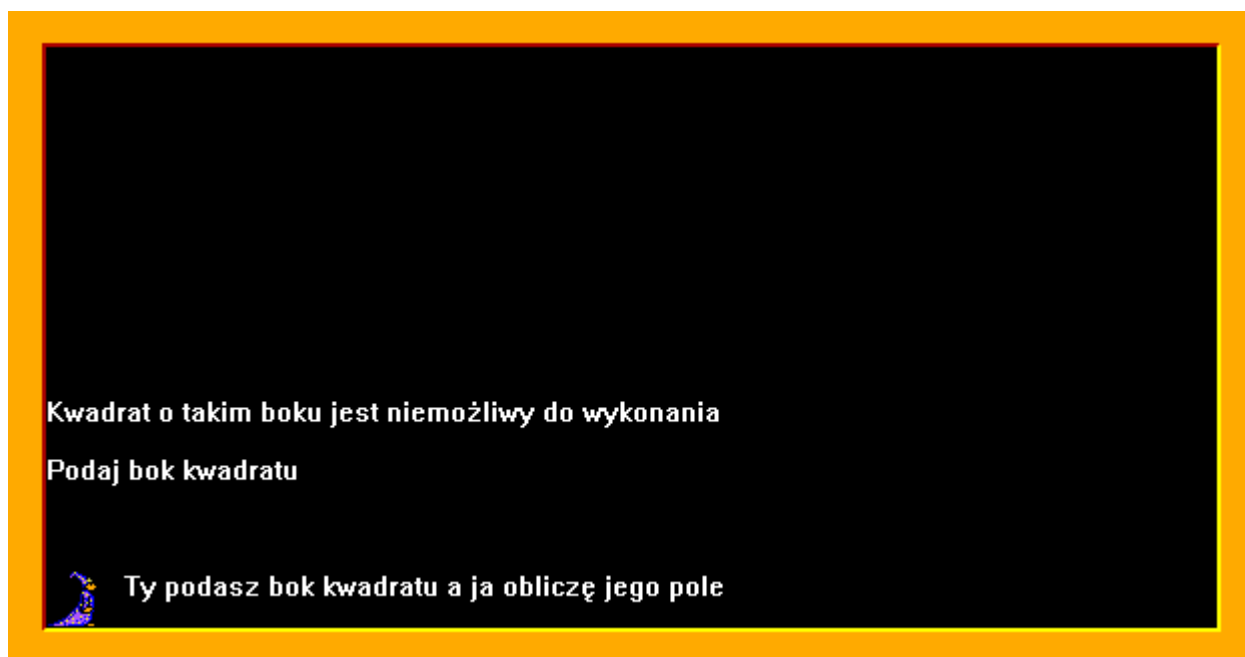
Program prosi o podanie boku kwadratu i oblicza jego pole. Zadanie jest proste więc je troszkę skomplikujemy. Gdy użytkownik poda liczbę ujemną jako bok kwadratu otrzyma informację zwrotną: „Kwadrat o takim boku jest niemożliwy do wykonania”

```
Ty podasz bok kwadratu a ja obliczę jego pole
Podaj bok kwadratu
a < 0
oczywiście musi być warunek
{ Kwadrat o takim boku jest niemożliwy do wykonania }
informacja zwrotna jesli warunek prawdziwy
ELSE
{ Pole wynosi:
  4 , 6
  jeśli warunek fałszywy pole obliczymy mnożąc bok razy bok
}
```

Efekt wywołania programu dla bok = 10

```
Pole wynosi:      100
Podaj bok kwadratu

Ty podasz bok kwadratu a ja obliczę jego pole
```



Zadanie 11

Jak to z Gaussem było?

Kim jest ten Gauss? Wybitny matematyk, fizyk, astronom. Ale tak jak Wy był też dzieckiem i uczniem i to nie zawsze grzecznym. Zdarzyło się, że za karę dostał pracochłonne zadanie: obliczyć sumę 200 kolejnych liczb naturalnych, od jedynki poczynając. Czyli miał do siebie dodawać $1+2+3+4+5+\dots+199+200$. Ku zdziwieniu nauczyciela uczeń bardzo szybko podał wynik i rozrabiał dalej. Gauss nie sumował kolejnych liczb lecz znalazł odpowiedni wzór, czyli algorytm, dzięki któremu łatwo obliczył sumę. Zauważył, że jeśli doda pierwszą liczbę i ostatnią, następnie drugą liczbę i przedostatnią i tak dalej uzyska te same wyniki, czyli $1+200$ i $2+199$ i $3+198$ zawsze daje 201. Szybko zauważył że dla 200 takich par jest 100, czyli połowa. Wystarczyło więc wykonać jedno działanie: $201 * 100 = 20100$. Jeżeli założymy że dodajemy do siebie kolejno n liczb naturalnych, czyli zaczynamy od 1 a kończymy na n. Uzyskamy dla liczb parzystych wzór $(1+n) * (n/2)$. Dla $n = 200$ wzór jest taki: $(1+200) * (200/2)$. Dla $n = 1000$ taki: $(1+1000) * 500$. Dla 5000: $(1+5000) * 2500$.

Dla liczb nieparzystych sytuacja jest podobna. Np. dla 101. Wzór jest taki: $(1+100) * (100/2) + 101$. Czyli stosujemy wzór dla liczb parzystych (ale dla $n-1$) i na końcu dodajemy liczbę n.

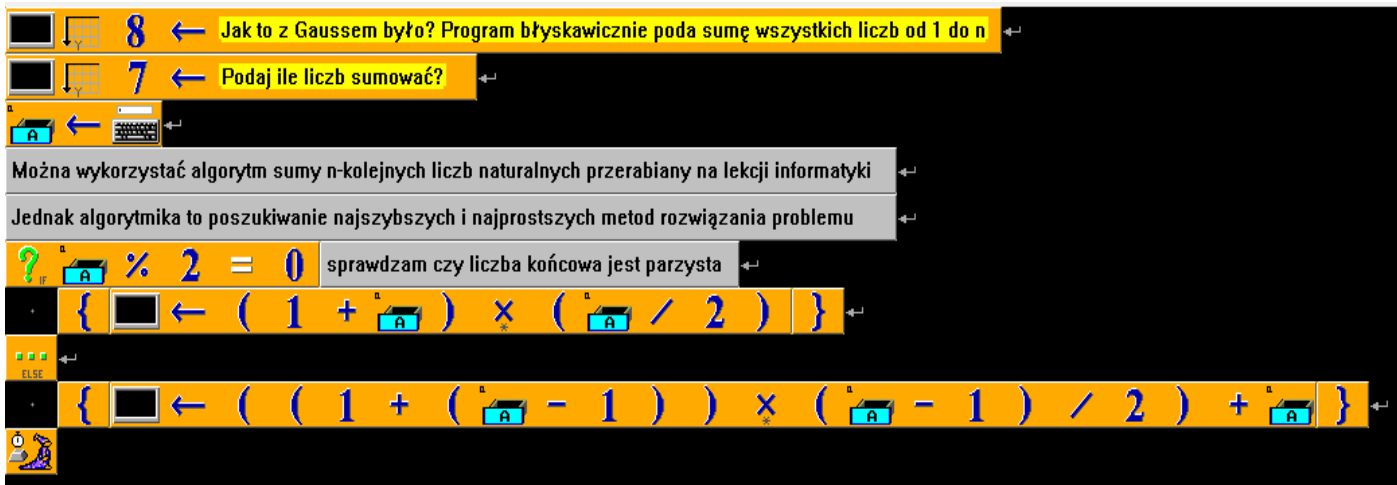
Zapamiętajcie

Dla parzystych np. 100

$$(1+100) * (100/2)$$

Dla nieparzystych np. 101

$$(1+100) * (100/2) + 101$$



To były tylko przykładowe metody rozwiązania zadań. Wasze mogą być inne co nie oznacza, że gorsze, a może wręcz lepsze. Zachęcam do własnych eksperymentów. W informatyce nie ma jednej słusznej drogi rozwiązania problemu, czyli znalezienia właściwego algorytmu - język programowania jest tylko narzędziem – a poznamy ich jeszcze kilka. Każda metoda, która prowadzi do celu jest dobra. Ale pamiętajcie o etapach rozwiązywania problemów (podręcznik strona 103 – punkt 7 **ocena efektywności przyjętej metody!!!** To dlatego Zadanie 6/127 zostało w 1/4 rozwiązania przerwane i zaproponowałem inne rozwiązanie).

POWODZENIA i jak najlepszych ocen na sprawdzianach