

## Sprawozdanie z informatyki nr. 1

## Treść zadanie

## Programownie część pierwsza

- 1.0 - Wyświetl napis powitalny, 2 wiersze poniżej ndk, cls, pożegnanie
- 1.1 - Wartość wyrażenia, 2 miejsca po przecinku:  $\frac{2*3+17}{9}$  (wynik 2, 56)
- 1.2 - Wartość wyrażenia, 3 miejsca po przecinku:  $\frac{4^2+2*4*7+7^2}{5+37\div 4}$  (wynik 8, 491)
- 1.3 - Powitanie, cls, pole pow. i objętość sześcianu o boku 5,875 cm (2mpp) (p=207,09; obj=202,78)
- 1.4 - Pole objętość i suma długości krawędzi sześcianu o boku 7.225 cm (ze stałą)  
 $P = 313, 20$ ;  $O = 377, 15$ ;  $S = 86, 70$
- 1.5 - Średnia arytmetyczna z 3 liczb - stałe (7,12,16) oraz napis NDK Średnia=11,67
- 1.6 - Pole, objętość i suma długości krawędzi prostopadłościanu o bokach podanych przez użytkownika (2mpp)
- 1.7 - Pole, objętość i suma długości krawędzi walca o promieniu i wysokości podanych przez użytkownika (pi jako stała)
- 1.8 - Średnia arytmetyczna z 3 liczb podanych przez użytkownika z jego imieniem.
- 1.9 - Pole prostokąta, ze sprawdzeniem danych (czy boki są większe od 0)
- 1.10 - Czy dwie liczby podane przez użytkownika są podzielne przez siebie – pierwsza przez drugą – uwaga na warunek podzielności
- 1.11 - Równanie liniowe w pełnej postaci ( $ax + b = c$ )
- 1.12 - Równanie  $ax^2 + bx + c = d$  kwadratowe lub liniowe – wyniki na dole ekranu
- 1.13 - Wyświetlanie maksymalnej liczby z trzech podanych liczb całkowitych ze sprawdzeniem poprawności danych
- 1.14 - Sprawdzenie możliwości skonstruowania trójkąta z trzech odcinków, których długości są losowymi liczbami rzeczywistymi, losowanymi z przedziału od <1,10> z jednym miejscem po przecinku (wyświetla liczby na górze i komunikat na dole ekranu)

## Proponowane rozwiązania

## 1.0

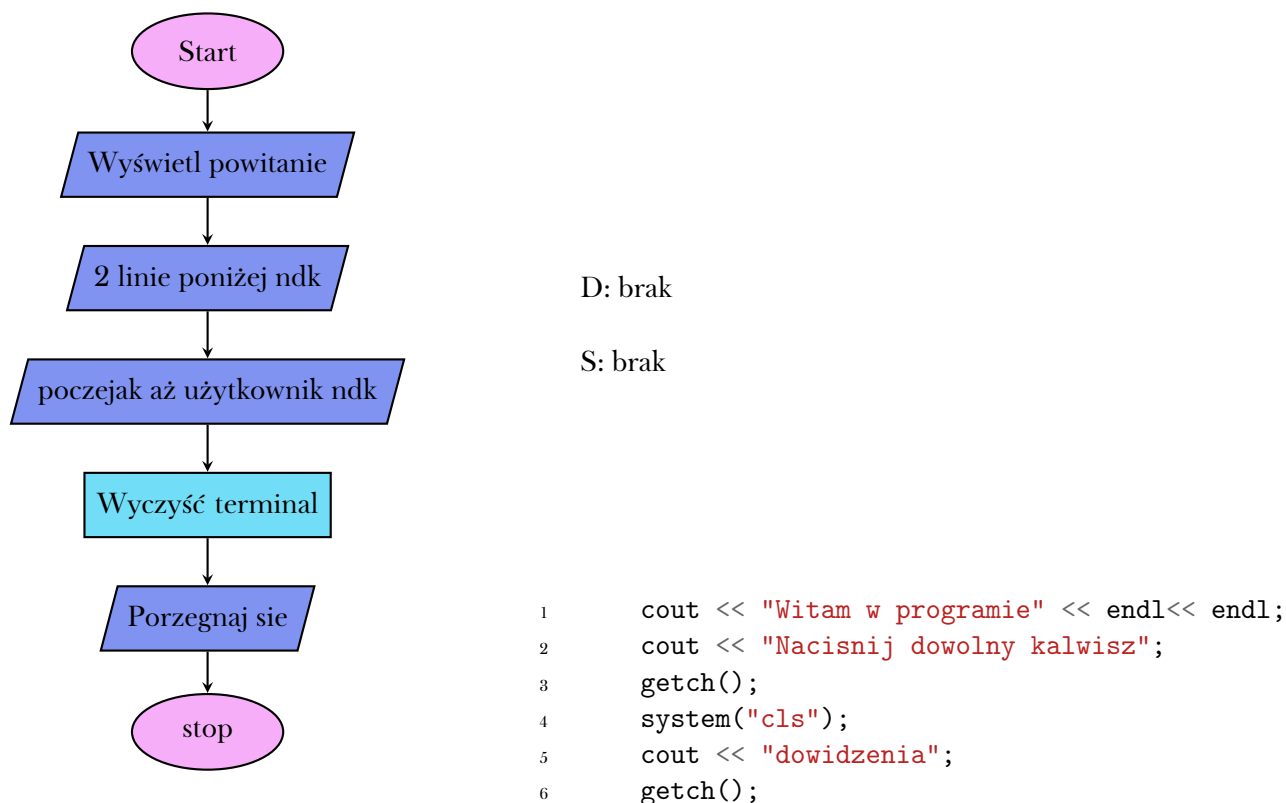


Figure 1: 1.0 flowchart

## 1.1

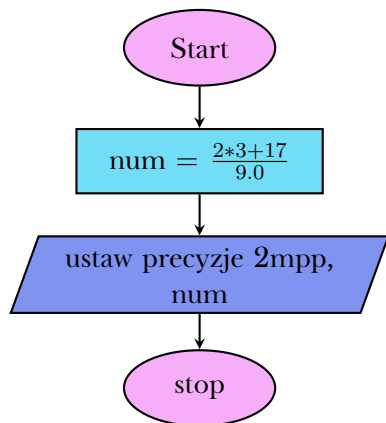


Figure 2: 1.1 flowchart

D:?

S: num Wartość wyrażenia  $\in 2,56$

```

1  double num = (2*3+17)/9.0;
2  cout << setprecision(3) << num;
  
```

## 1.2

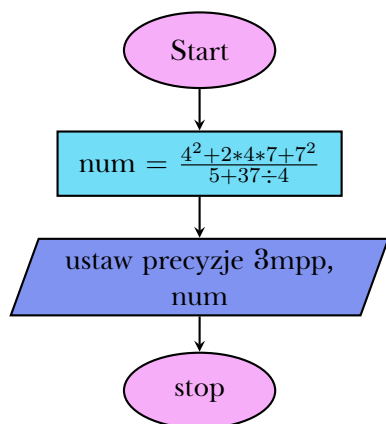


Figure 3: 1.2 flowchart

D:?

S: num wartość wyrażenia  $\in 8,491$

```

1  float num = (pow(4,2)+2*4*7+pow(7,2))/(5+37/4.0);
2  cout << setprecision(4) << num;
  
```

## 1.3

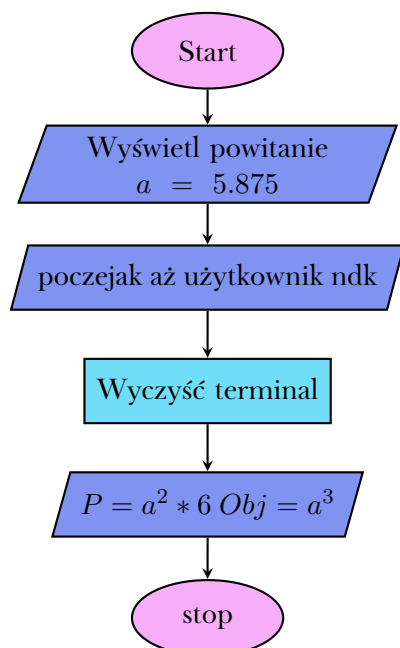


Figure 4: 1.3 flowchart

D: a bok  $\in 5.875$

S: P, Obj wymiary  $\in \mathbb{R}$

```

1  string pow ("witam w programie");
2  char* pow_arr = new char[pow.length()];
3  strcpy(pow_arr, pow.c_str());
4  for(n=0;n<=pow.length()-1;n++){
5      cout << pow_arr[n];
6      Sleep(250);}
7  Sleep(2000);
8  system("cls");
9  float bok = 5.875;
10 float pole = 6.0*bok*bok;
11 float obj = 1.0*bok*bok*bok;
12 cout<<setprecision(3)<<"P="<<pole<<"Obj="<< obj;
  
```

## 1.4

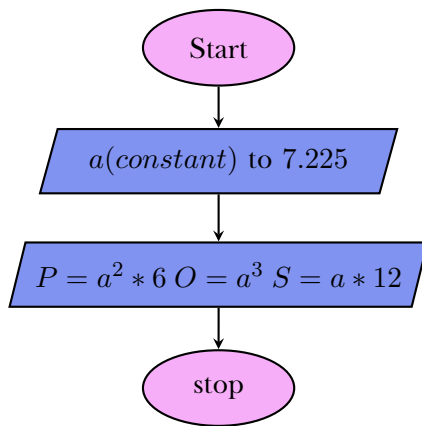


Figure 5: 1.4 flowchart

D:  $a \text{ bok} \in \{7.225\}$

S:  $P, O, S \text{ wymiary} \in \{313, 20; 377, 15; 86, 70\}$

```

1 cout <<fixed<< setprecision(2)<<\
2 "dla boku:"<<a << endl <<\
3 "P="<< a*a*6<< endl <<\
4 "O=" << a*a*a << endl<<\
5 "S="<< a*12;
  
```

## 1.5

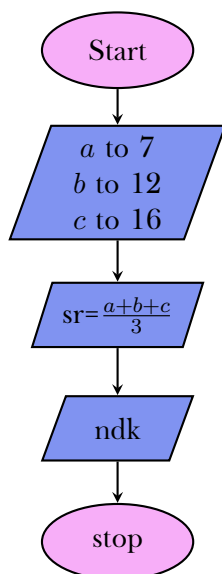


Figure 6: 1.5 flowchart

D:  $a, b, c \text{ parametry} \in \{7; 12; 16\}$

S:  $sr \text{ średnia parametrów} \in \{11, 67\}$

```

1 double sr = (a+b+c)/3.0;
2 cout << fixed <<setprecision(2)<<\
3 "Srednia: " << sr << endl
  
```

## 1.6

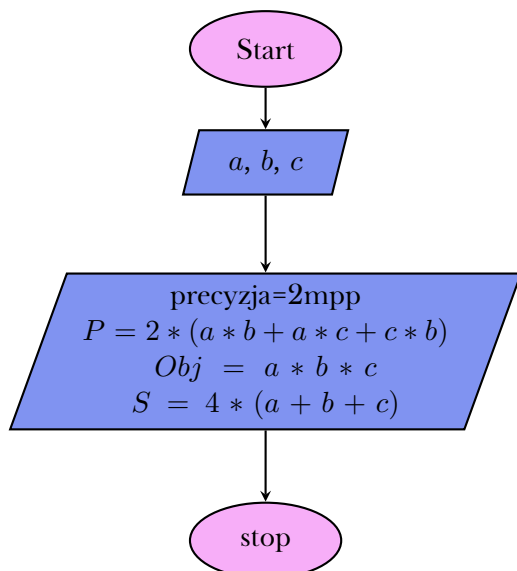


Figure 7: 1.6 flowchart

D:  $a, b, c \text{ parametry} \in \mathbb{R} > 0$

S:  $P, Obj, S \text{ wymiary} \in \mathbb{R}$

```

1 float a;
2 float b;
3 float c;
4 cout<<"Prosze podac bok a:"; cin >> a;
5 cout <<"Prosze podac bok b:"; cin >> b;
6 cout <<"Prosze podac bok c:"; cin >> c;
7 cout<<fixed<<setprecision(2)<<\
8 "Objetosc jest równa="<<a*b*c<<endl<<\
9 "pole jest równe="<<2*(a*b+a*c+b*c)<<endl<<\
10 "Suma dlugosci krawedzi to="<<4*(a+b+c);
  
```

## 1.7

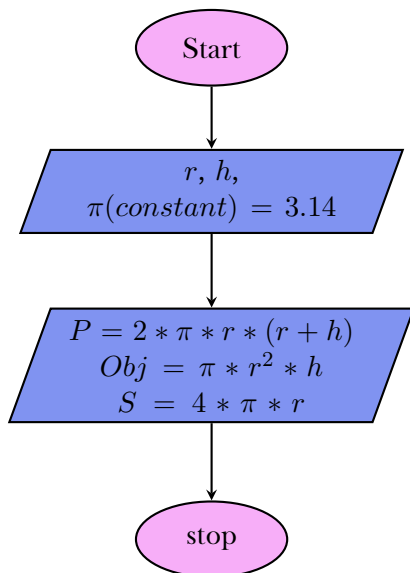


Figure 8: 1.7 flowchart

D:  $r, h$  parametry  $\in \mathbb{R} > 0$   
 S:  $P, Obj, S$  wymiary  $\in \mathbb{R}$

```

1 float r;
2 float h;
3 const float pi=3.14;
4 cout<<"Proszę podać promień:"; cin >> r;
5 cout <<"Proszę podać wysokość:"; cin >> h;
6 cout<<fixed<<setprecision(2)<<\
7 "Objętość jest równa="<<pi*r*r*h<<endl<<\
8 "pole jest równe="<<2*pi*r*(r+h)<<endl<<\
9 "Suma długości krawędzi to="<<4*pi*r;
  
```

## 1.8

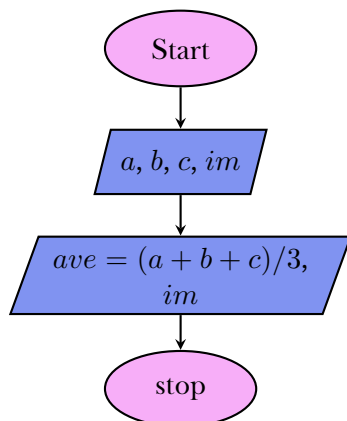


Figure 9: 1.8 flowchart

D:  $a, b, c, im$ ; Liczy i imię użytkownika  $\in \mathbb{R}$   
 S:  $ave$  średnia  $\in \mathbb{R}$

```

1 float a;
2 float b;
3 float c;
4 string name;
5 cout<<"Podaj imię użytkownika:"; cin >> name;
6 cout<<"Proszę podać pierwszą liczbę:"; cin >> a;
7 cout<<"Proszę podać drugą liczbę:"; cin >> b;
8 cout<<"Proszę podać trzecią liczbę:"; cin >> c;
9 float ave = (a+b+c)/3;
10 cout<<"Średnia jest równa ="<<ave<<endl;
11 cout<<"Imię użytkownika to: "<<name<<endl;
  
```

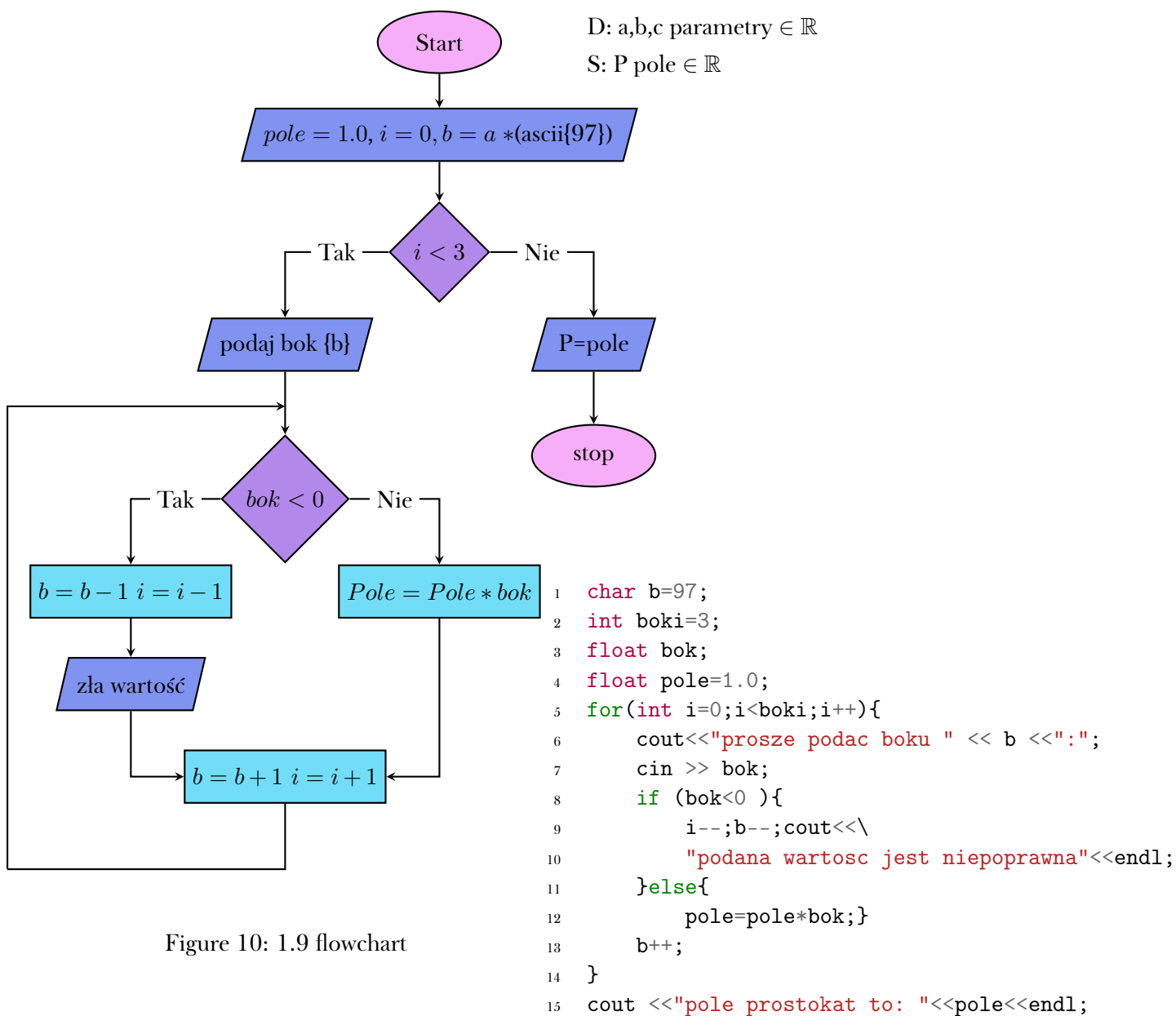
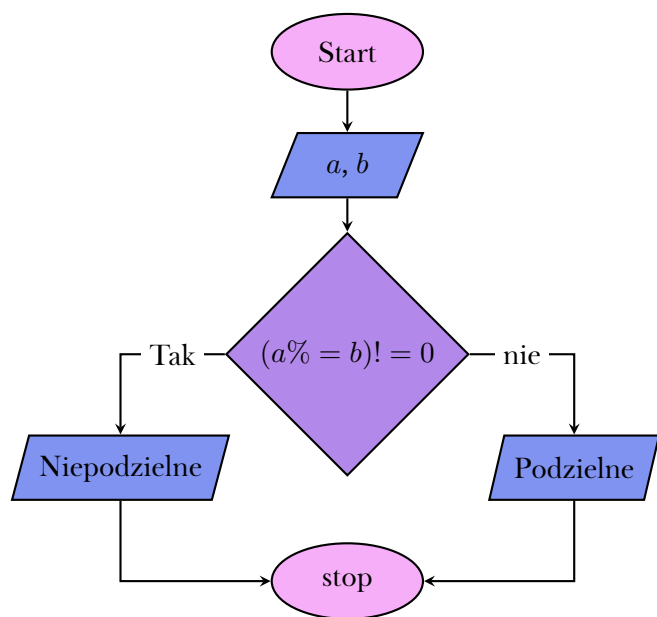


Figure 10: 1.9 flowchart

## 1.10



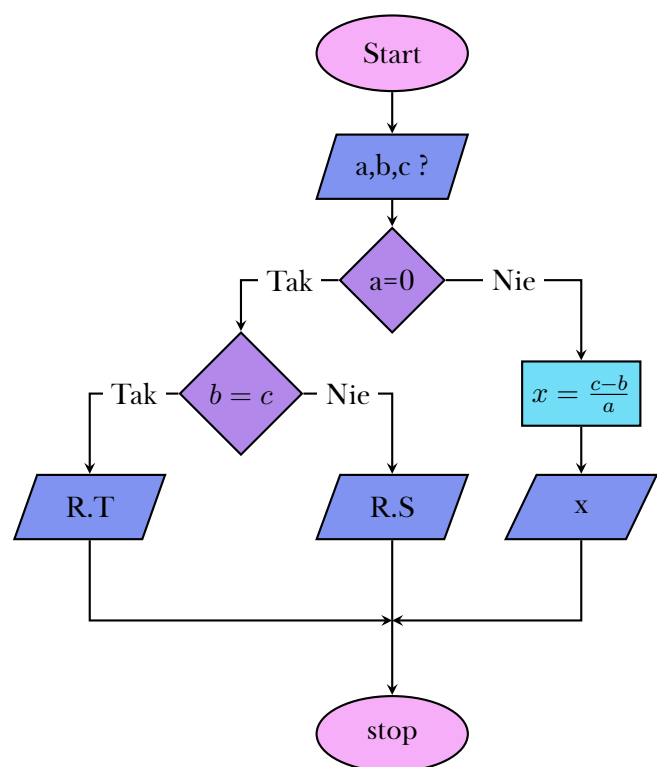
D:  $a, b$  liczby  $\in \mathbb{R}$  czy są w rzeczywistości  
 S:  $x$  podzielność  $\in$  prawda/fałsz

```

1  int num1;
2  int num2;
3  cout<<"prosze podac pierwszą liczbę:"; cin >> num1;
4  cout<<"prosze podac drugą liczbę:"; cin >> num2;
5  num1 %= num2;
6  if(num1!=0){cout<<"liczby są niepodzielne"<< endl;
7  }else{cout<<"liczby są podzielne"<< endl;}
  
```

Figure 11: 1.10 flowchart

## 1.11



Równanie:  $ax + b = c$

D:  $a, b, c$  wyznaczniki  $\in \mathbb{R}$

S:  $x$  rozwiązanie równania  $\in \mathbb{R}$

```

1  float a;
2  float b;
3  float c;
4  cout<<"prosze podać a:"; cin>>a;
5  cout<<"prosze podać b:"; cin>>b;
6  cout<<"prosze podać c:"; cin>>c;
7  if(a==0){
8      if(b==c){cout<<"Równanie tożsamościowe"<<endl;
9      }else{cout<<"równanie sprzeczne"<< endl;}}
10 else{a=(c-b)/a;cout<<"x jest równy: "<< a<<endl;}
  
```

Figure 12: 1.11 flowchart

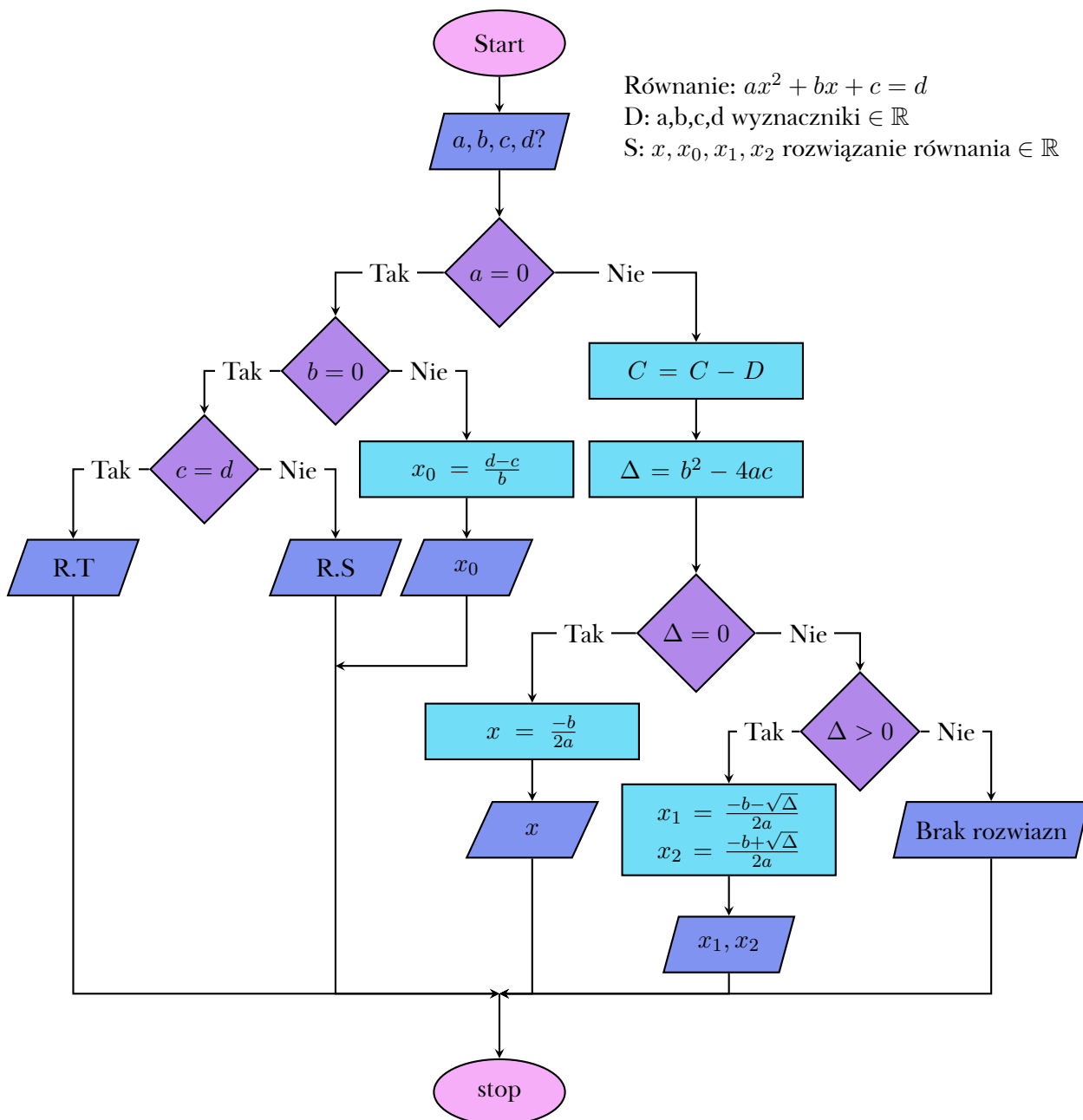
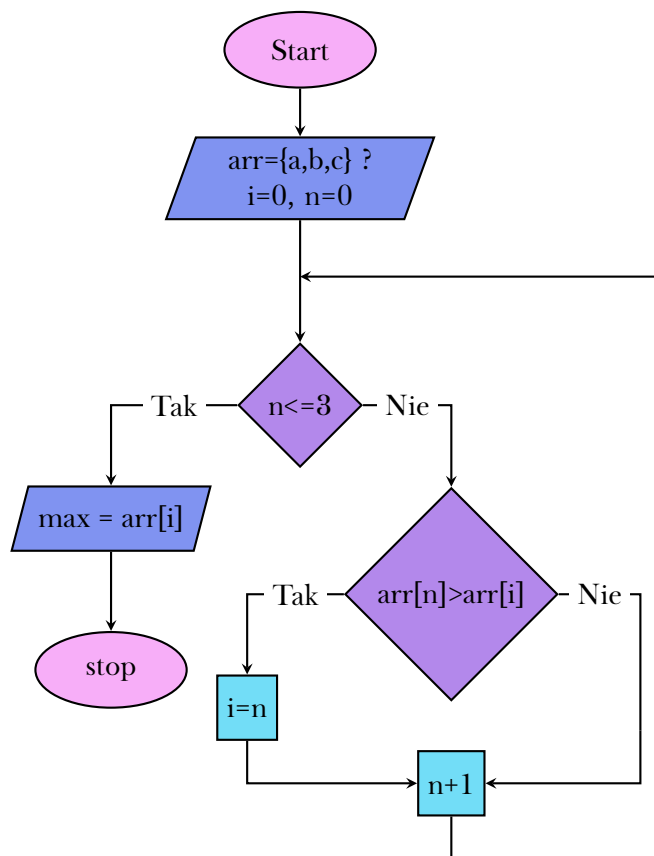


Figure 13: 1.12 flowchart

```

1  float a; cout<<"prosze podać a:"; cin>>a; float b; cout<<"prosze podać b:"; cin>>b;
2  float c; cout<<"prosze podać c:"; cin>>c; float d; cout<<"prosze podać d:"; cin>>d;
3  float delta; gotoxy(0,25);
4  if(a==0){
5      if(b==0){
6          if(c==d){cout<<"Równanie tożsamościowe"<<endl;}
7          else{cout<<"równanie sprzeczne"<< endl;}}
8      else{b=(d-c)/b;cout<<"x jest równy: "<<b<<endl;}
9  }else{ c=c-d;
10     float delta=pow(b,2)-(4.0*a*c);
11     if(delta==0){a=(-b)/(2*a));cout<<"x jest równy: "<< a <<endl;
12     }else{
13         if(delta>0){cout<<"x1 jest równy: "<< ((-b-sqrt(delta))/(2*a))<<endl;
14             cout<<"x2 jest równy: "<<((-b+sqrt(delta))/(2*a))<<endl;
15         }else{cout<<"brak rozwiązn"<<endl;}}
  
```

## 1.13



D: a,b,c liczby  $\in \mathbb{N}_0$   
 S: max największa liczba  $\in \mathbb{N}_0$

```

1  int n = 3;
2  int t;
3  int arr[n];
4  int i=0;
5  while(i<n)
6  {   cout << "prosze podać " << i+1 << " liczbe:";
7      cin >> t;
8      arr[i]={t};
9      i++;}
10 n = 0;
11 i = 0;
12 while(n<sizeof(arr)/sizeof(arr[0]))
13 {   if(arr[n]>arr[i]){i=n;}
14     n++;}
15 cout << "max number is: " << arr[i];
  
```

Figure 14: 1.13 flowchart

## 1.14



D: brak  
 S: możliwość zbydowania trójkąta {tak, nie}

```

1      int n = 3;
2      float a[n];
3      float b;
4
5      for(int i=0;i<n;i++)
6      {a[i-1]={ceil((rand()/ double(RAND_MAX))*100
7
8      for(int i=0;i<n;i++)
9      {for(int j=0;j<n;j++)
10     {if(a[i-1]<a[j-1])
11      {b=a[i-1];a[i-1]=a[j-1];a[j-1]=b;}}}
12
13     gotoxy(0,0);
14     print_arr(a,n);
15
16     gotoxy(0,28);
17
18     if(a[-1]+a[0]>a[1]){cout<<"tak"<<endl;}
19     else{cout<<"nie"<<endl;}
  
```

Figure 15: 1.14 flowchart



*Programy użyte do wykonania zadań*

---

*Wnioski i uwagi*

---

Zadanie mi się bardzo podobało i nie mam żadnych uwag.