

# Zadanie 1 trojmian.c

Aleksander Kłak, prowadzący Łukasz Janiec

4 listopada 2020

## Spis treści

1	Przygotowanie do zajęć	1
2	Kompilacja	1
3	Kod programu	1
4	Działanie programu oraz testy	2
5	Wnioski	3

## 1 Przygotowanie do zajęć

Przygotowaniem do zajęć był algorytm zapisany w schemacie blokowym oraz wykonanie testów. Na GitLab umieściłem zarówno algorytm jak i przetestowanie algorytmu wraz z jego wynikami. Oba pliki znajdują się w repozytorium `aklak_podstawy_programowania/Z1/przygotowanie`

## 2 Kompilacja

Program kompiluje się w każdym środowisku jakiego zastosowałem. Stosowałem Visual Studio Code oraz internetowe zintegrowane środowisko programistyczne Repl.it. Do skompilowania programu potrzebna była poprawa paru niewielkich błędów. Teraz program działa bez zarzutu.

## 3 Kod programu

Listing 1: C++

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     float a,b,c,delta;
6
7     printf("Program oblicza pierwiastki rownania w postaci\n");
8     printf("      2\n");
9     printf("  a x + b x +c =0\n");
10    printf("Podaj wartosc a:");
11    scanf("%f",&a);
12    printf("Podaj wartosc b:");
13    scanf("%f",&b);
```

```

14 printf("Podaj wartosc c:");
15 scanf("%f",&c);
16
17 if (a==0.0) /*przypadek rownania liniowego */
18     if (b!=0.0)
19         printf("Jest to rownanie liniowe o rozwiazaniu x=%f\n",-c/b);
20     else if (c==0.0) /* oraz a=b==0.0 */
21         printf("Rozwiazaniem jest dowolne x\n");
22     else /* a=b==0.0 != c */
23         printf("Brak rozwiazan\n");
24 else { /*przypadek rownania kwadratowego */
25     delta=pow(b,2)-4.0*a*c;
26     if (delta<0)
27         printf("Brak rozwiazan rzeczywistych\n");
28     else /* delta>=0 */
29         if (delta>0)
30             printf("Rozwiazaniem sa x1=%f i x2=%f\n",(-b-sqrt(delta))/(2*a),(-b+sqrt(delta))/(2*a));
31         else
32             printf("Rozwiazaniem sa x1=x2=%f\n",-b/(2*a));
33     }
34 return 0;
35 }

```

## 4 Działanie programu oraz testy

Program po wspomnianych wyżej poprawkach buduje się i uruchamia bezbłędnie. Wprowadzanie danych, po czym program wykonuje procesy matematyczne. Na sprzęcie prywatnym procesy wykonują się prędko oraz poprawnie, jednak z niewielką dokładnością, ale sporym zakresem. Dzieje się tak z powodu zadeklarowania zmiennych jako *float*. Dopiero przy wartości  $2^{150}$  wystąpił błąd i program pokazał "nan" zamiast liczb.

Tabela 1: Testy

a	b	c	$x_1$	$x_2$	$x_0$
0	0	0	dowolne x		
0	0	1	brak rozwiązań		
0	1	0			0
1	0	0			0
0	1	1			1
1	1	0	1	0	
1	0	1	brak rozwiązań		
1	1	1	brak rozwiązań		
1	2	1			-1
65536	1	-1	-0,003914	0,003899	
2147483648	1	-1	-0,000022	0,000022	
4294967296	1	-1	-0,000015	0,00015	
$2^{150}$	1	-1	nan	-nan	

## 5 Wnioski

Najważniejszym wnioskiem z zadania jest poprawne zapamiętanie i stosowanie składni języka programowania. Każdy język ma swoje specyficzne elementy składni bez których programowanie jest niemożliwe. Kolejnym wnioskiem może być umiejętność korzystania z kompilatora. Dobry kompilator może pomóc w rozwiązywaniu błędów. Błędy i ich rozwiązania są szeroko dostępne w internecie, należy jedynie umieć do nich dotrzeć. Ostatnim wnioskiem jest uprzednie przemyślenie typu zmiennej, tak aby jej rozmiar i jej zakres zgadzały się z tym, co jest od zmiennej wymagane i do czego zostanie użyta.