

Wojciech Rogowski

**Wybór zadań w C++
z rozwiązaniami**

ZSEI Bielsko-Biała 2009

1. Systemy liczbowe

Zadanie 1.

Znajdź liczby z przedziału n do m , które są podzielne przez 5. Liczby n , m określające przedział są wprowadzone z klawiatury.

Dane:

n (początek zakresu) liczba wprowadzona z klawiatury typu integer.
 m (koniec zakresu) liczba wprowadzona z klawiatury typu integer.

Rozwiązanie problemu: C++ posiada dwa operatory dzielenia:

`%` reszta z dzielenia,

`/` część całkowita z dzielenia (gdy zmienna została zadeklarowana jako np. integer)

Przy rozwiązaniu tego zadania postępujemy podobnie jak w **Zadaniu 3**. Z lekcji matematyki wiemy, że liczba jest podzielna przez 5 wtedy i tylko wtedy gdy ostatnią jej cyfrą jest 0 lub 5. Jeśli ten warunek jest spełniony to taką liczbę wypisujemy. Rozwiązując ten problem musimy zastosować pętlę:

for do sprawdzenia czy ostatnia cyfra liczba z podanego przedziału spełnia zadany warunek (znamy ilość liczb).

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienne n , m , i pom ,
K_02	Wyczyść ekran,
K_03	wyświetl komunikat <i>Podaj początek zakresu:</i>
K_04	wprowadź z klawiatury wartość zmiennej n ,
K_05	wyświetl komunikat <i>Podaj koniec zakresu:</i>
K_06	wprowadź z klawiatury wartość zmiennej m ,
K_07	dla $n \leq i \leq m$ zwiększ n o 1 i wykonuj kroki K_08, K_09 i K_10,
K_08	$pom \leftarrow$ reszta z dzielenia $i:10$,
K_09	jeżeli $i > 9$ i $pom = 0$ wypisz i .
K_10	w przeciwnym wypadku jeżeli $pom = 5$ wypisz i .
K_11	wczytaj znak z klawiatury.

Listing 001:

```
/*001*/      #include <iostream.h>
/*002*/      #include <conio.h>
/*003*/      void main(){
/*004*/          int n, m, pom;
/*005*/          clrscr();
/*006*/          cout<<"Podaj poczatek zakresu: ";
/*007*/          cin>>n;
/*008*/          cout<<" Podaj koniec zakresu: ";
/*009*/          cin>>m;
/*010*/          for(int i=n; i<=m; i++){
/*011*/              pom=i%10;
/*012*/              if((i>9) &&(pom==0)) cout<<i<<endl;
/*013*/              else if(pom==5)cout<<i<<endl;
/*014*/              }
/*015*/          getch();
/*016*/      }
```

Zadanie 2.

Znajdź liczby z przedziału n do m , które są podzielne przez 2. Liczby n , m określające przedział są wprowadzone z klawiatury.

Dane:

n (początek zakresu) liczba wprowadzona z klawiatury typu integer.
 m (koniec zakresu) liczba wprowadzona z klawiatury typu integer.

Rozwiązanie problemu:

Przy rozwiązaniu tego zadania postępujemy podobnie jak w *Zadaniu 1*. Z lekcji matematyki wiemy, że liczba jest podzielna przez 2 wtedy i tylko wtedy gdy ostatnią jej cyfrą jest 0, 2, 4, 6, lub 8. Jeśli ten warunek jest spełniony to taką liczbę wypisujemy. Rozwiązując ten problem musimy zastosować pętlę:

for do sprawdzenia czy ostatnia cyfra liczby z podanego przedziału spełnia zadany warunek (znamy ilość liczb).

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienne n , m , i pom ,
K_02	Wyczyść ekran,
K_03	wyświetl komunikat <i>Podaj początek zakresu:</i>
K_04	wprowadź z klawiatury wartość zmiennej n ,
K_05	wyświetl komunikat <i>Podaj koniec zakresu:</i>
K_06	wprowadź z klawiatury wartość zmiennej m ,
K_07	dla $n \leq i \leq m$ zwiększ n o 1 i wykonuj kroki K_08, K_09 i K_10,
K_08	$pom \leftarrow$ reszta z dzielenia $i:10$,
K_09	jeżeli $i > 9$ i $pom = 0$ wypisz i .
K_10	w przeciwnym wypadku jeżeli $pom = 2$ lub $pom = 4$ lub $pom = 6$ lub $pom = 8$ wypisz i .
K_11	wczytaj znak z klawiatury.

Listing_002:

```
/*001*/ #include <iostream.h>
/*002*/ #include <conio.h>
/*003*/ void main(){
/*004*/     int n, m, pom;
/*005*/     clrscr();
/*006*/     cout<<"Podaj poczatek zakresu: ";
/*007*/     cin>>n;
/*008*/     cout<<" Podaj koniec zakresu: ";
/*009*/     cin>>m;
/*010*/     for(int i=n; i<=m; i++){
/*011*/         pom=i%10;
/*012*/         if((i>9) &&(pom==0)) cout<<i<<endl;
/*013*/         else if((pom==2) || (pom==4) || (pom==6) || (pom==8))
/*014*/             cout<<i<<endl;
/*015*/         }
/*016*/     getch();
/*017*/ }
```

Zadanie 3.

Znajdź sumę cyfr liczby w wprowadzonej z klawiatury.

Dane:

w liczba wprowadzona z klawiatury typu integer.

Rozwiązanie problemu:

Przyjmijmy, że z klawiatury wprowadzono liczbę **123**. Łatwo obliczyć, że suma cyfr tej liczby wynosi 6. W tym celu zsumowaliśmy ilość jedności, dziesiątek i setek. Czyli tak naprawdę dzieliliśmy tak długo naszą liczbę przez **10**, aż otrzymaliśmy wartość mniejszą od 1.

Tak więc w wyniku działania: **123%10** otrzymamy resztę z dzielenia równą **3**,

A w wyniku działania: **123/10** otrzymamy część całkowitą z dzielenia czyli **12**,

Powtarzając w następnym kroku te operacje otrzymamy odpowiednio:

12%10 otrzymamy resztę z dzielenia równą **2**,

12/10 otrzymamy część całkowitą z dzielenia czyli **1**.

Sumując wszystkie reszty z dzielenia otrzymamy rozwiązanie zadania.

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienną <i>liczba</i> i zainicjalizujemy zmienną <i>suma</i> $\leftarrow 0$,
K_02	Wyczyść ekran,
K_03	wyświetl komunikat <i>Podaj liczbę:</i>
K_04	wprowadź z klawiatury wartość zmiennej <i>liczba</i> ,
K_05	dopóki <i>liczba</i> jest większa od 0 powtarzaj krok K_06 i K_07,
K_06	$suma \leftarrow suma + \text{reszta z dzielenia } liczba:10$,
K_07	$liczba \leftarrow \text{część całkowita z dzielenia } liczba:10$,
K_08	wczytaj znak z klawiatury.

Listing_003:

```
/*001*/ #include <iostream.h>
/*002*/ #include <conio.h>
/*003*/ void main(){
/*004*/     int suma=0, liczba;
/*005*/     clrscr();
/*006*/     cout<<"Podaj liczbe: ";
/*007*/     cin>>liczba;
/*008*/     while(liczba){
/*009*/         suma+=liczba%10;
/*010*/         liczba/=10;
/*011*/     }
/*012*/     cout<<"Obliczona suma: "<<suma<<endl;
/*013*/     getch();
/*014*/ }
```

Zadanie 4

Znajdź liczby z przedziału n do m spełniające następujący warunek: iloczyn wszystkich cyfr danej liczby jest równy w . Przedział n , m i warunek w są wprowadzone z klawiatury.

Dane:

n (początek zakresu) liczba wprowadzona z klawiatury typu integer.
 m (koniec zakresu) liczba wprowadzona z klawiatury typu integer.
 w (warunek) liczba wprowadzona z klawiatury typu integer.

Rozwiązanie problemu:

Przy rozwiązaniu tego zadania postępujemy podobnie jak w **Zadaniu 3**, z tą jednak różnicą, że obliczamy iloczyny cyfr każdej liczby z podanego przedziału i sprawdzamy czy spełniają one warunek w . Jeśli tak to taką liczbę wypisujemy. Rozwiązując ten problem musimy zastosować dwie pętle:

for do sprawdzenia czy liczba z podanego przedziału spełnia zadany warunek (znamy ich ilość),
while do obliczenia iloczynu cyfr sprawdzanej liczby (ilość cyfr zmienia się w czasie obliczeń).

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienne l , n , m , w i il ,
K_02	Wyczyść ekran,
K_03	wyświetl komunikat Podaj początek zakresu:
K_04	wprowadź z klawiatury wartość zmiennej n ,
K_05	wyświetl komunikat Podaj koniec zakresu:
K_06	wprowadź z klawiatury wartość zmiennej m ,
K_07	wyświetl komunikat Podaj warunek:
K_08	wprowadź z klawiatury wartość zmiennej w ,
K_09	dla $n \leq i \leq m$ zwiększ n o 1 i wykonuj kroki K_10 , K_11 , K_12 , i K_15
K_10	$l \leftarrow i$,
K_11	$il \leftarrow 1$,
K_12	dopóki l jest większa od 0 powtarzaj krok K_13 i K_14 ,
K_13	$il \leftarrow il * \text{reszta z dzielenia } l:10$,
K_14	$l \leftarrow \text{część całkowita z dzielenia } l:10$,
K_15	jeżeli $il = w$ wypisz i .
K_16	wczytaj znak z klawiatury.

Listing_004:

```
/*001*/ #include <iostream.h>
/*002*/ #include <conio.h>
/*003*/ void main(){
/*004*/     int l, n, m, w, il;
/*005*/     clrscr();
/*006*/     cout<<"Podaj początek zakresu: ";
/*008*/     cin>>n;
/*009*/     cout<<" Podaj koniec zakresu: ";
/*010*/     cin>>m;
/*011*/     cout<<"          Podaj warunek: ";
/*012*/     cin>>w;
/*013*/     for(int i=n; i<=m; i++){
/*014*/         l=i;
```

```

/*015*/      il=1;
/*016*/      while(l){
/*017*/          il*=l%10;
/*018*/          l/=10;
/*019*/      }
/*020*/      if(il==w) cout<<i<<endl;
/*021*/      }
/*022*/      getch();
/*023*/      }

```

Zadanie 5.

Znajdź liczby z przedziału n do m , które są podzielne przez 3. Liczby n , m określające przedział są wprowadzone z klawiatury.

Dane:

n (początek zakresu) liczba wprowadzona z klawiatury typu integer.
 m (koniec zakresu) liczba wprowadzona z klawiatury typu integer.

Rozwiązanie problemu:

Przy rozwiązaniu tego zadania postępujemy podobnie jak w **Zadaniu 3**. Z lekcji matematyki wiemy, że liczba jest podzielna przez 3 wtedy i tylko wtedy gdy suma jej cyfr jest podzielna przez 3. Jeśli ten warunek jest spełniony to taką liczbę wypisujemy. Rozwiązując ten problem musimy zastosować dwie pętle:

for do sprawdzenia czy liczba z podanego przedziału spełnia zadany warunek (znamy ich ilość),
while do obliczenia iloczynu cyfr sprawdzanej liczby (ilość cyfr zmienia się w czasie obliczeń).

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienne l , n , m , i sum ,
K_02	Wyczyść ekran,
K_03	wyświetl komunikat Podaj początek zakresu:
K_04	wprowadź z klawiatury wartość zmiennej n ,
K_05	wyświetl komunikat Podaj koniec zakresu:
K_06	wprowadź z klawiatury wartość zmiennej m ,
K_07	dla $n \leq i \leq m$ zwiększ n o 1 i wykonuj kroki K_08 , K_09 , K_10 , i K_13
K_08	$l \leftarrow i$,
K_09	$sum \leftarrow 0$,
K_10	dopóki l jest większa od 0 powtarzaj krok K_13 i K_14 ,
K_11	$sum \leftarrow sum + \text{reszta z dzielenia } l:10$,
K_12	$l \leftarrow \text{część całkowita z dzielenia } l:10$,
K_13	jeżeli reszta z dzielenia $sum:3 = 0$ wypisz i .
K_14	wczytaj znak z klawiatury.

Listing 005:

```

/*001*/      #include <iostream.h>
/*002*/      #include <conio.h>
/*003*/      void main(){
/*004*/          int l, n, m, sum;
/*005*/          clrscr();
/*006*/          cout<<"Podaj poczatek zakresu: ";
/*007*/          cin>>n;
/*008*/          cout<<" Podaj koniec zakresu: ";

```

```

/*009*/      cin>>m;
/*010*/      for(int i=n; i<=m; i++){
/*011*/          l=i;
/*012*/          sum=0;
/*013*/          while(l){
/*014*/              sum+=l%10;
/*015*/              l/=10;
/*016*/          }
/*017*/          if(!(sum%3)) cout<<i<<endl;
/*018*/      }
/*019*/      getch();
/*020*/      }

```

Zadanie 6.

Zamień wprowadzoną z klawiatury liczbę binarną n na dziesiętną.

Dane:

n liczba wprowadzona z klawiatury typu *char*.

Rozwiązanie problemu:

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienną <i>string</i> jako tablicę 20-sto elementową typu char,
K_02	utwórz prototyp funkcji pot(), która zwraca wartość typu integer
K_03	wyczyść ekran,
K_04	wyświetl komunikat <i>Podaj liczbę binarną:</i>
K_05	wprowadź z klawiatury wartość zmiennej <i>string</i> ,
K_06	zainicjalizuj zmienne $suma \leftarrow 0$ i $i \leftarrow strlen(string)$,
K_07	dla $i \geq n >= 0$ zmniejsz n o 1 i wykonuj krok K_08,
K_08	jeżeli $string[n-1] = '1'$ oblicz wyrażenie $sum += pot(i-n)^1$,
K_09	wyświetl komunikat <i>Po zamianie:</i> i wartość zmiennej <i>sum</i> ,
K_10	wczytaj znak z klawiatury.
Algorytm funkcji <i>pot()</i>	
K_11	jeżeli pot(0) wtedy zwróć 1
K_12	w przeciwnym wypadku zwróć $e * pot(--e)^2$

Listing 006:

```

/*001*/      #include <iostream.h>
/*002*/      #include <string.h>
/*003*/      #include <conio.h>
/*004*/      char string[20];
/*005*/      int pot(int e);
/*006*/      void main(){
/*007*/          clrscr();
/*008*/          cout<<"Podaj liczbę binarną: ";
/*009*/          cin>>string;
/*010*/          int sum=0, i=strlen(string);
/*011*/          for(int n=i; n>0; n--)
/*012*/              if(string[n-1]=='1') sum+=pot(i-n);
/*013*/          cout<<"Po zamianie: "<<sum<<endl;
/*014*/          getch();
/*015*/      }
/*016*/      int pot(int e){

```

¹ pot() jest funkcją zwracającą wartość obliczonej potęgi

² Do obliczenia potęgi zastosowano rekurencję

```

/*017*/      int p;
/*018*/      if(!e) p=1;
/*019*/      else p=2*pot(--e);
/*020*/      return p;
/*021*/      }

```

Zadanie 7.

Niech będzie dana liczba dziesiętna n typu *long*, podaj szesnastkową postać sumy jej cyfr.

Dane:

n liczba wprowadzona z klawiatury typu *long*.

Szukane:

s suma cyfr liczby n typu *int*,

h szesnastkowy zapis liczby s .

Rozwiązanie problemu:

Algorytm:

Krok	Opis
K_01	zadeklaruj zmienną h jako tablicę 6-cio elementową i c typu char,
K_02	utwórz prototyp funkcji <code>sumaCyfr(long l)</code> , która zwróci wartość typu integer,
K_03	utwórz prototyp funkcji <code>dec_to_hex(int l)</code> , która zwróci wartość typu char,
Algorytm funkcji <code>main()</code>	
K_04	zainicjuj zmienną $c \leftarrow 0$ typu char,
K_05	zadeklaruj zmienną $liczba$,
K_06	Dopóki $c \neq 27$ powtarzaj kroki:
K_07	wyczyść ekran,
K_08	wyświetl komunikat „Podaj liczbę:” 9 .. 14,
K_09	wprowadź z klawiatury wartość zmiennej $liczba$,
K_10	zainicjuj zmienną s typu integer i przypisz jej wartość zwracaną przez funkcję <code>sumaCyfr()</code> ,
K_11	wyświetl komunikat „suma=” i wartość s
K_12	wyświetl komunikat „hex=”, i wartość zwracaną przez funkcję <code>dec_to_hex()</code> ,
K_13	wyświetl komunikat „----- Esc - koniec -----”,
K_14	zmiennej c przypisz kod klawisza,
Algorytm funkcji <code>sumaCyfr()</code>	
K_15	

Listing 007:

```

/*001*/      #include <iostream.h>
/*002*/      #include <conio.h>
/*003*/      #include <string.h>
/*004*/      //=====
/*005*/      //=
/*006*/      //= Zadanie 7
/*007*/      //=
/*008*/      //=====
/*009*/      char h[6];
/*010*/      int sumaCyfr(long l);
/*011*/      char *dec_to_hex(int l);
/*012*/      void main(){
/*013*/          char c=0;
/*014*/          long liczba;

```



```

/*015*/         while(c!=27){
/*016*/             clrscr();
/*017*/             cout<<"Podaj liczbe: ";
/*018*/             cin>>liczba;
/*019*/             int s=sumaCyfr(liczba);
/*020*/             cout<<"\nsuma= "<<s<<endl;
/*021*/             cout<<"\n hex= "<<dec_to_hex(s)<<endl;
/*022*/             cout<<"\n----- Esc - koniec ----- ";
/*023*/             c=getch();
/*024*/         }
/*025*/     }
/*026*/     int sumaCyfr(long l){
/*027*/         int s=0;
/*028*/         while(l){
/*029*/             s+=l%10;
/*030*/             l/=10;
/*031*/         }
/*032*/         return s;
/*033*/     }
/*034*/     char *dec_to_hex(int l){
/*035*/         int p,k=-1,i=1;
/*036*/         while(i){
/*037*/             i/=16;
/*038*/             ++k;
/*039*/         }
/*040*/         while(l){
/*041*/             p=l%16;
/*042*/             if(p<10) h[k-i]=char(48+p);
/*043*/             else h[k-i]=char(55+p);
/*044*/             l/=16;
/*045*/             i++;
/*046*/         }
/*047*/         h[i]=0;
/*048*/     }

```