Εργαστήριο
Εισαγωγή στον Προγραμματισμό

Εργαστήριο 03

Δομές επανάληψης

Βασιλόπουλος Διονύσης

Ε.ΔΙ.Π. Τμήματος Πληροφορικής & Τηλεπικοινωνιών

ΤΜΗΜΑ
ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

4ο Εργαστήριο

Άθροισμα αριθμών 1 .. 100

```
#include <stdio.h>

int sum() {

int i=0;
int temp=0;

while (i<100) {

    temp+=++i; //Equal to the following 2 instructions
  //i=i+1;
  //temp=temp+i;

}

return temp;

}

int main()
{

printf("The sum is: %d \n", sum());

return 0;

}
```

Όταν χρησιμοποιείτε μεταβλητές για υπολογισμούς σχεδόν πάντα χρειάζεται να την αρχικοποιείτε (συνήθως στο 0).

4ο Εργαστήριο

Άθροισμα αριθμών 1, 1/4, 1/9 .. 1/10000

```
#include <stdio.h>

float basel() {

//int i=0;
float temp=0;

for (int i=1;i<=100;i++) {

    //temp+=(float)1/(i*i); //Solution 1
    temp+=1.0/(i*i);        //Solution 2
    printf("temp: %f, i: %d\n", temp, i);

}

return temp;

}

int main()
{

printf("The sum is: %f \n", basel());

return 0;

}
```

temp+=1/(i*i);

Σε αυτή την περίπτωση το temp είναι πάντα 0 εκτός της 1ης επανάληψης. Οπότε το άθροισμα γίνεται 1.

Λάθος και το:
temp+=(float)(1/(i*i));
Γιατί;

https://www.tutorialspoint.com/cprogramming/c_type_casting.htm

4ο Εργαστήριο

Άθροισμα αριθμών 1, 1/4, 1/9 .. 1/10000, …..

```c
#include <stdio.h>

float basel() {

float temp=0;

for (int i=1;i<=100;i++) {

    //temp+=(float)1/(i*i);
    temp+=1.0/(i*i);
    printf("temp: %f, i: %d\n", temp, i);

}

return temp;

}

int main()
{

printf("The sum is: %f \n", basel());

return 0;

}
```

Basel problem
->1.634984

```c
#include <stdio.h>

float basel(int times) {

float temp=0;

for (int i=1;i<=times;i++) {

    temp+=1.0/(i*i);
    printf("temp: %f, i: %d\n", temp, i);

}

return temp;

}

int main()
{

printf("The sum is: %f \n", basel(200));

return 0;

}
```

Basel problem
->1.644934

Πιο γενική συνάρτηση από την αρχική

4ο Εργαστήριο

Pi (https://www.imdb.com/title/tt0138704/)

```
#include <stdio.h>
#include <math.h>

float pi_approx() {

int i=0;
float temp=0;

do {
    i++;
    temp+=1.0/(i*i);
    printf("temp: %f, i: %d\n", temp, i);

} while (i<100);

return sqrt(6*temp);

}

int main()
{

printf("The sum is: %f \n", pi_approx());

return 0;

}
```

4ο Εργαστήριο

Pi (with precision) - float

```
#include <stdio.h>
#include <math.h>

float pi_approx_detail() {

int i=0;
double temp=0;
double new_term;

do {
    i++;

    //new_term=1.0/(i*i);

    new_term=(float)1.0/((float)i*(float)i);

    temp+=new_term;
    printf("temp: %f, i: %d, term: %.20f \n", temp, i, new_term);

} while (new_term>=1e-15);

return sqrt(6*temp);

}

int main()
{

printf("The sum is: %f \n", pi_approx_detail());

return 0;

}
```

Stops at i=46341
Term= -0.00000000046566228651
Sum= 3.141393
overflow INT_MAX = 2,147,483,647
returns to negative

Stops at i=31622778
Term=0.00000000000000100000
Sum= 3.141393

Target: 3.14159265

4ο Εργαστήριο

Pi (with precision) - double

```
#include <stdio.h>
#include <math.h>

float pi_approx_detail() {

int i=0;
double temp=0;
double new_term;

do {
    i++;

    //new_term=1.0/(i*i);

    new_term=1.0/((double)i*(double)i);

    temp+=new_term;
    printf("temp: %f, i: %d, term: %.20f \n", temp, i, new_term);

} while (new_term>=1e-15);

return sqrt(6*temp);

}

int main()
{

printf("The sum is: %f \n", pi_approx_detail());

return 0;

}
```

Stops at i=46341
Term= -0.00000000046566229193
Sum= 3.141572

Stops at i=31622777
Term=0.00000000000000100000
Sum= 3.141593

Target: 3.14159265

https://codefinity.com/blog/Float-vs-Double

4ο Εργαστήριο

Leibniz sequence

$$S_L = \sum_{i=0}^{\infty} \frac{(-1)^i}{2i+1} \rightarrow \frac{\pi}{4} \rightarrow 0{,}7853981125$$

Θέλουμε ο χρήστης να εισάγει το n, και να τυπώνεται το αποτέλεσμα

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>
```

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>

int main(){

//Ζητάω από το χρήστη ένα αριθμό ακέραιο
//Διαβάζω τον ακέραιο αριθμό από το πληκτρολόγιο
//Υπολογίζω το άθροισμα
//Τυπώνω το αποτέλεσμα

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>

int main(){

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);                    //Error στο compile

//Υπολογίζω το άθροισμα
//Τυπώνω το αποτέλεσμα

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>

int main(){

int terms;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

//Υπολογίζω το άθροισμα
//Τυπώνω το αποτέλεσμα

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>

int main(){

int terms;
double lei_sum=0.0;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

//Υπολογίζω το άθροισμα
printf("The sum is: %10.8f \n", lei_sum);

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence

```
#include <stdio.h>

//dummy function
double leibniz(int terms) {

double temp=0.0;


return temp;

}
```

```
#include <stdio.h>

int main(){

int terms;
double lei_sum=0.0;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

lei_sum=leibniz(terms);
printf("The sum is: %10.8f \n", lei_sum);

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence: $S_L = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \rightarrow \frac{\pi}{4} \rightarrow 0{,}7853981125$

```
#include <stdio.h>

double leibniz(int terms) {

int i=0;
double temp=0.0;

for(i=0;i<terms;i++){

}

return temp;

}
```

```
#include <stdio.h>

int main(){

int terms;
double lei_sum=0.0;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

lei_sum=leibniz(terms);
printf("The sum is: %10.8f \n", lei_sum);

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence: $S_L = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \rightarrow \frac{\pi}{4} \rightarrow 0{,}7853981125$

```
#include <stdio.h>
#include <math.h>

double leibniz(int terms) {
double new_value=0.0;

int i=0;
double temp=0.0;

for(i=0;i<terms;i++){
new_value=pow(-1, i)/(2*(double)i+1);
}
return temp;

}
```

```
#include <stdio.h>

int main(){

int terms;
double lei_sum=0.0;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

lei_sum=leibniz(terms);
printf("The sum is: %10.8f \n", lei_sum);

return 0;
}
```

4ο Εργαστήριο

Leibniz sequence: $S_L = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \rightarrow \frac{\pi}{4} \rightarrow 0{,}7853981125$

```
#include <stdio.h>
#include <math.h>

double leibniz(int terms) {
double new_value=0.0;

int i=0;
double temp=0.0;

for(i=0;i<terms;i++){
new_value=pow(-1, i)/(2*(double)i+1);
temp+=new_value;
}
return temp;

}
```

```
int main(){

int terms;
double lei_sum=0.0;

printf("Please enter the number of terms to use: ");
scanf("%d", &terms);

lei_sum=leibniz(terms);
printf("The sum is: %10.8f \n", lei_sum);

return 0;
}
```

4ο Εργαστήριο

Compile

```
gcc -o seq_leibniz seq_leibniz.c -lm
```

4ο Εργαστήριο

limit

Let the sequence: $S \quad = \sum_{n=0}^{\infty} \left(-\frac{1}{2}\right)^n \rightarrow \frac{2}{3} \rightarrow 0{,}6666666$

or

$s = \frac{a}{1-r}$, όπου α ο πρώτος όρος (1), και r ο κοινός όρος ($\frac{1}{2}$)

$$Limit\ Sequence = \ S_{limit} = \sum_{n=0}^{m} \left(-\frac{1}{2}\right)^n, \text{while } \left(-\frac{1}{2}\right)^n > limit$$

4ο Εργαστήριο

srand – rand functions

https://www.tutorialspoint.com/c_standard_library/c_function_rand.htm
https://www.tutorialspoint.com/c_standard_library/c_function_srand.htm

srand:

Reproducibility: If you use the same seed, you'll get the same sequence of numbers. This can be useful for debugging or when you need reproducible results.

Single Call: Generally, srand() is only called once in a program. Calling srand() multiple times with the same seed will reset the sequence each time, which might not be desirable.

4ο Εργαστήριο

Math functions

ceil(x): Επιστρέφει ως double τον πλησιέστερο ακέραιο που είναι μεγαλύτερος ή ίσος του x
https://www.w3schools.com/c/ref_math_ceil.php

trunc(x): Επιστρέφει ως double το ακέραιο μέρος του x
https://www.w3schools.com/c/ref_math_trunc.php

floor(x): Επιστρέφει ως double τον πλησιέστερο ακέραιο που είναι μικρότερος ή ίσος του x
https://www.w3schools.com/c/ref_math_floor.php

https://www.w3schools.com/c/c_ref_math.php

4ο Εργαστήριο

Libraries

https://www.tutorialspoint.com/c_standard_library/stdio_h.htm

https://www.tutorialspoint.com/c_standard_library/stdlib_h.htm

https://www.tutorialspoint.com/c_standard_library/math_h.htm