**C Programming**

1. Program to print hello:

#include <stdio.h>

int main() {

printf("Hello World");

return 0;

}

The sections are: Main function, curly braces, end with semicolon, inbuilt functions must be included.

Header files are connected to the c compiler through IDE by set paths variables during installations. These path variables are defined during installation and they connect to the default directory. They are part of the include file in the default directory of the compiler.

1. Program and to take runtime input of two numbers and print it
2. #include <stdio.h>
3. void main(){
4. int x,y;
5. printf(">>");
6. scanf("%d",&x);
7. printf(">>");
8. scanf("%d",&y);
9. printf("the numbers are %d and %d",x,y);
10. }
11. Write a program to give three numbers as input:

#include <stdio.h>

int main() {

int b1,b2,b3,product;

printf("Enter number 1:");

scanf("%d",&b1);

printf("Enter number 2:");

scanf("%d",&b2);

printf("Enter number 3:");

scanf("%d",&b3);

product=b1\*b2\*b3;

printf("/nThe product is:%d",product);

}

**HackerRank Questions:**

1. Hello World:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

char s[100];

scanf("%[^\n]%\*c", &s);

/\* Enter your code here. Read input from STDIN. Print output to STDOUT \*/

printf("Hello, World! \n%s",&s);

return 0;

}

1. Playing with Characters:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

/\* Enter your code here. Read input from STDIN. Print output to STDOUT \*/

char ch;

char s[20];

char sen[100];

scanf("%c",&ch);

scanf("%s\n",s);

scanf("%[^\n]%\*c",sen);

printf("%c\n",ch);

printf("%s\n",s);

printf("%s",sen);

return 0;

}

1. Sum and Difference:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

int a,b,sum,sub;

float x,y,sums,subs;

scanf("%d %d",&a,&b);

scanf("%f %f",&x,&y);

sum=a+b;

sub=a-b;

sums=x+y;

subs=x-y;

printf("%d %d\n",sum,sub);

printf("%0.1f %0.1f",sums,subs);

return 0;

}

1. Functions in C:

#include <stdio.h>

/\*

Add `int max\_of\_four(int a, int b, int c, int d)` here.

\*/

int max\_of\_four(int a,int b, int c, int d) {

if (a>b && a>c && a>d){

return a;

}

else if(b>a && b>c && b>d){

return b;

}

else if(c>a && c>b && c>d){

return c;

}

else{

return d;

}

}

int main() {

int a, b, c, d;

scanf("%d %d %d %d", &a, &b, &c, &d);

int ans = max\_of\_four(a, b, c, d);

printf("%d", ans);

return 0;

}

1. Pointers in C:

#include <stdio.h>

#include <stdlib.h>

void update(int \*a,int \*b) {

// Complete this function

int sum, diff;

sum= \*a + \*b;

diff=abs(\*a-\*b);

\*a=sum;

\*b=diff;

}

int main() {

int a, b;

int \*pa = &a, \*pb = &b;

scanf("%d %d", &a, &b);

update(pa, pb);

printf("%d\n%d", a, b);

return 0;

}

1. Conditional Statements:

#include <assert.h>

#include <limits.h>

#include <math.h>

#include <stdbool.h>

#include <stddef.h>

#include <stdint.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

char\* readline();

int main()

{

char\* n\_endptr;

char\* n\_str = readline();

int n = strtol(n\_str, &n\_endptr, 10);

if (n\_endptr == n\_str || \*n\_endptr != '\0') { exit(EXIT\_FAILURE); }

// Write Your Code Here

if (n==1){

printf("one");

}

else if(n==2){

printf("two");

}

else if(n==3){

printf("three");

}

else if(n==4){

printf("four");

}

else if(n==5){

printf("five");

}

else if(n==6){

printf("six");

}

else if(n==7){

printf("seven");

}

else if(n==8){

printf("eight");

}

else if(n==9){

printf("nine");

}

else if(n>9){

printf("Greater than 9");

}

return 0;

}

char\* readline() {

size\_t alloc\_length = 1024;

size\_t data\_length = 0;

char\* data = malloc(alloc\_length);

while (true) {

char\* cursor = data + data\_length;

char\* line = fgets(cursor, alloc\_length - data\_length, stdin);

if (!line) { break; }

data\_length += strlen(cursor);

if (data\_length < alloc\_length - 1 || data[data\_length - 1] == '\n') { break; }

size\_t new\_length = alloc\_length << 1;

data = realloc(data, new\_length);

if (!data) { break; }

alloc\_length = new\_length;

}

if (data[data\_length - 1] == '\n') {

data[data\_length - 1] = '\0';

}

data = realloc(data, data\_length);

return data;

}