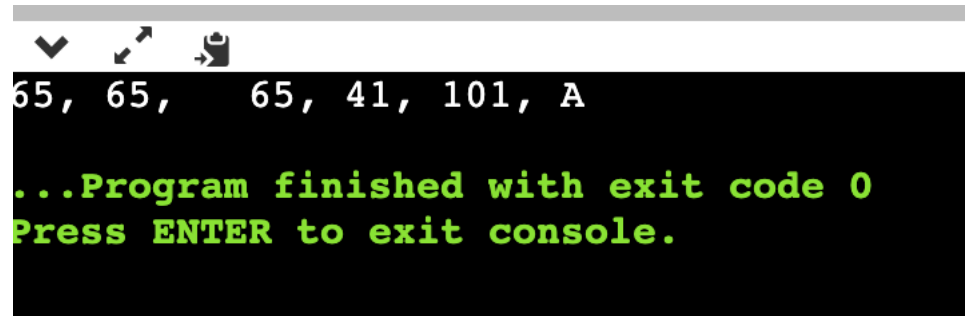


1.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char * * argv)
{
    int a = 65;
    printf("%d, %d, %4d, %x, %o, %c", 65, 65, 65, 65, 65, (char)65);
}
```



```
65, 65, 65, 41, 101, A
...Program finished with exit code 0
Press ENTER to exit console.
```

2.

```
#include <stdio.h>
int main(int argc, char * * argv){
    int a;
    char b;
    float c;
    double d;
    printf("size of int %lu\n", sizeof(a));
    printf("size of char %lu\n", sizeof(b));
    printf("size of float %lu\n", sizeof(c));
    printf("size of double %lu\n", sizeof(d));
}

(base) yulongwang@YulongdeMBP lec2 % gcc -c size.c
(base) yulongwang@YulongdeMBP lec2 % gcc -o size size.o
(base) yulongwang@YulongdeMBP lec2 % ./size
size of int 4
size of char 1
size of float 4
size of double 8
```

The size here is the number of bytes for each datatype.

3.

```
/* Minimum and maximum values a `signed int' can hold. */
# define INT_MIN    (-INT_MAX - 1)
# define INT_MAX    2147483647
So -2147483648 to 2147483647
```

#### 4.

```
// first.c
#include <stdio.h>
#include <stdlib.h>
#define MAX 256
#define PUSH 1
#define POP 0
#define LIST 2
int main(int argc, char* argv[])
{
    int stack[MAX];
    int size = 0;
    int val;
    int iChoice;
    int iNRead;
    int result;
    int newElement;
    int read;

    /* Processing loop */
    printf("Choice (1=add, 0=remove, 2=list): ");
    iNRead = scanf("%d", &iChoice);
    while(iNRead == 1)
    {
        switch(iChoice)
        {
            case PUSH:
                printf("Value to add: ");
                iNRead = scanf("%d", &newElement);
                if(size < MAX){
                    stack[size] = newElement;
                    size++;
                }
                // Read the element, add it to the stack
                break;
            case POP:
                result = stack[size-1];
                stack[size] = 0;
                size--;
                printf("Value popped: %d", result);
                // Print out the last element and remove it.
                break;
            case LIST:
                for(int i=0;i<size;i++){
```

```

        printf("%d\n",stack[i]);
    }
    break;
}
printf("Choice (1=add, 0=remove, 2=list): ");
iNRead = scanf("%d", &iChoice);
}
return EXIT_SUCCESS;
}

```

```

[(base) yulongwang@YulongdeMBP Lec2src % gcc -c playStack.c
[(base) yulongwang@YulongdeMBP Lec2src % gcc -o playStack playStack.o
[(base) yulongwang@YulongdeMBP Lec2src % ./playStack
Choice (1=add, 0=remove, 2=list): 1
Value to add:
1
Choice (1=add, 0=remove, 2=list): 1
Value to add: 2
Choice (1=add, 0=remove, 2=list): 1
Value to add: 3
Choice (1=add, 0=remove, 2=list): 2
1
2
3
Choice (1=add, 0=remove, 2=list): 0
Value popped: 3Choice (1=add, 0=remove, 2=list): 2
1
2
Choice (1=add, 0=remove, 2=list): █

```