



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA

Guía de Ejercicios – Punteros
Algoritmos y Estructura de datos

Curso Ing. Gustavo Schmidt

Ejercicio 1: Codificar

- 1 - Declarar las variables enteras a y b asignar 10 a a y 20 a b
- 2 - Declarar p y q como punteros a int
- 3 - Asignar a p la direccion de a
- 4 - Emitir el valor de a y su direccion
- 5 - Emitir la direccion contenida en p, el valor contenido en la direccion contenida en p, y la direccion de p
- 6 - Asignar a q la direccion de a
- 7 - Emitir la direccion contenida en q, el valor contenido en la direccion contenida en q y la direccion de q
- 8 - Sumar 1 a la variable a usando el puntero p
- 9- Asignar a q la direccion de b
- 10- Asignar a la dirección almacenada en p el contenido de la dirección almacenada en q
- 11-Emitir la direccion contenida en p, el valor contenido en la direccion contenida en p, y la direccion de p
- 12- Asignar a p la dirección almacenada en q
- 13-Emitir la direccion contenida en p, el valor contenido en la direccion contenida en p, y la direccion de p

Ejercicio 2: codificar

- 1-Declarar a y b como int. a=1, b=2
- 2-Declarar c y d como char c= 'A' . d= 'E'
- 3- Declarar p puntero a int. Asignarle a p la direccion de a.
- 4- Modificar el contenido de a usando p, asignarle 87.
- 5- Declarar q puntero a char.
- 6- Asignarle a q la direccion de c
- 7- Modificar el contenido de c, asignandole el de d, usando el puntero q.
- 8- Emitir el contenido de p, su dirección, el contenido de la dirección almacenada en p. Idem para q.
- 9- Asignar (previo casting) a q la direccion de p
- 10- Declarar pp como puntero a puntero a int. asignarle la direccion de p
- 11- A través de pp, hacer que p apunte a b
- 12- Emitir el contenido de pp, su dirección , el contenido de la dirección almacenada en pp, y **pp

Ejercicio 3:

Hacer una funcion para intercambiar los valores de 2 enteros, ya sea con referencias como con punteros. Serian 4 versiones en total. Hacer el main del caso de uso.

Ejercicio 4: Hacer el seguimiento

```
/* Punteros
 * Ejercicio Tipo Parcial N1
 */

#include <stdlib.h>
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
  Datos:
    '@' es 64
    'A' es 65
    ...
 */

int main(){

    Pint A, C, J, F;
    Pint* B;
    Pchar D, E;
    char G;
    int H;

    H = 64;
    G = 'C';
    A = new int;
    F = new int;
    (*A) = 70;
    cout << H << G << (*A) << endl;
    B = &A;
    (*F) = H;
    cout << (**B) << (*A) << endl;
    D = (Pchar)F;
    E = (Pchar)(*B);
    C = (Pint)D;
```

```

J = (Pint)E;
cout << (*D) << (*E) << (**B) << endl;
(*C) = (*C) - 62;
(*J) = (*J) - (*J);

if ((*F) == H) {
    cout << G << H << (*E) << endl;
}

while ((*C) != 0) {
    cout << (*A) << (*C) << endl;
    (*C) = (*C) - 1;
    (*E) = 'B';
}

delete A;
delete F;

return 0;
}

/*
Resultado:

    64C70
    7070
    @F70
    02
    661

*/

```

Ejercicio 5: Hacer el seguimiento

```

/* Punteros
 * Ejercicio Tipo Parcial N2 - Poder Legislativo (publicado en internet)
 */

#include <stdlib.h>
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*

```

```
Datos que pueden ser utiles: El caracter '?' es igual al entero 160.
                                El caracter '?' es igual al entero 130.
                                El caracter '?' es igual al entero 164.
                                El caracter '?' es igual al entero 165.
                                El caracter '@' es igual al entero 64.

*/

int main(){

    Pint A, C, F;
    Pint* B;
    Pchar D, E;
    char G;
    int H;

    H = 164;
    G = '?';
    A = new int;
    F = new int;
    (*A) = 160;
    cout << H << G << (*A) << endl;
    B = &A;
    (*F) = (*A) + H -159;
    (**B) = (*F);
    cout << (**B) << (*A) << endl;
    D = (Pchar)F;
    E = (Pchar)A;
    C = (Pint)D;
    cout << (*D) << (*C) << (**B) << endl;
    (*C) = (*C) - 163;

    if ((*D) == G) {
        cout << G << H << (*E) << endl;
    }

    while ((*C) != 0) {
        cout << (*E) << (*C) << endl;
        (*C) = (*C) - 1;
        (*F) = (*F) - 1;
    }

    (*C) = (*C) + 160;
    cout << (*F) << endl;

    delete A;
    delete F;

    return 0;
}
```

```
}  
  
/*  
    Resultado:  
  
        64 160  
        165165  
        165165  
        2  
        160  
*/
```

Ejercicio 6:

```
/* Punteros  
 * Ejercicio Tipo Parcial N 3  
 */  
  
#include <stdlib.h>  
#include <iostream>  
  
using namespace std;  
  
typedef int* Pint;  
typedef char* Pchar;  
  
/*  
    Datos:  
    '@' es 64  
    'A' es 65  
    ...  
*/  
  
int main(){  
  
    Pint A, C, F;  
    Pint* B;  
    Pchar D, E;  
    char G;  
    int H;  
  
    H = 66;  
    G = 'D';  
    A = new int;  
    F = new int;  
    (*A) = 64;  
    cout << H << G << (*A) << endl;
```

```

B = &F;
(*F) = (*A) + H - 62;
(**B) = (*F) - 2;
cout << (*F) << (*A) << (**B) << endl;
D = (Pchar)F;
E = (Pchar)A;
C = (*B);
cout << (*D) << (*C) << (*E) << endl;
(*C) = (*C) - 63;

if ((*F) == H) {
    cout << G << H << (*E) << endl;
}

while ((*C) > 0) {
    cout << (*E) << (*C) << endl;
    (*C) = (*C) - 1;
    (*F) = (*F) - 1;
    (*A) = 70;
}

if (**B == (*C)) {
    cout << (*E) << endl;
}

delete A;
delete F;

return 0;
}

/*
Resultado:
66D64
666466
B66@
@3
F1
F
*/

```

Ejercicio 7: hacer el seguimiento

```

/* Punteros
 * Ejercicio Tipo Parcial N 4
 */

#include <stdlib.h>

```

```
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
    Datos:
    '@' es 64
    'A' es 65
    ...
*/

int main(){

    Pint A, C, F;
    Pint* B;
    Pchar D, E;
    char G;
    int H;

    H = 66;
    G = 'D';
    A = new int;
    F = new int;
    (*A) = 64;
    B = &F;
    (**B) = (*A) - 63;
    cout << H << G << (*F)<< endl;

    (*F) = (*A) + H - 63;
    cout << (*F) << (*A) << (**B) << endl;

    D = (Pchar)F;
    E = (Pchar)A;
    C = (Pint)D;
    cout << (*D) << (*C) << (*E) << endl;

    (*C) = (*C) - 66;
    H++;
    if ((*F) == H) {
        cout << G << H << (*E) << endl;
    }

    while ((*C) >= 0) {
        cout << (*E) << (*C) << endl;
    }
}
```



```
    (*F) = (*F) - 1;
    (*C) = (*C) - 1;
}

delete A;
delete F;

return 0;
}

/*
Resultado:
66 D 1
67 64 67
C 67 @
@ 1
*/
```

Ejercicio 8: hacer el seguimiento

```
/* Punteros
 * Ejercicio Tipo Parcial N 5
 */

#include <stdlib.h>
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
Datos:
'@' es 64
'A' es 65
...
*/

int main(){

    Pint A, C, F;
    Pint* B;
    Pchar D, E;
    char G;
    int H;
```

```
H = 70;
G = 'C';
F = new int;
(*F) = H;
C = new int;
A = C;
(*A) = 67;
H++;

cout << (*C) << (*A) << (*F) << endl;

B = &A;
D = (Pchar)F;
E = (Pchar)(*B);
(**B) = (*A) - 63;

if ( (*E) != G ) {
    cout << (*A) << (*D) << (*C) << endl;
}

(*A) = (*A) - (*C) + 66;
cout << (*E) << (*F) << G << endl;

while ((*C) > 0) {

    (*E) = 'E';
    (*A) = (*F) - (*C);

    cout << (**B) << endl;

    (*C)--;
}

delete A;
delete F;

return 0;
}

/*
Resultado:
        676770
        4F4
        B70C
        1
*/
```

Ejercicio 9: Hacer el seguimiento

```
/* Punteros
 * Ejercicio Tipo Parcial N6
 */

#include <stdlib.h>
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
  Datos:
    '@' es 64
    'A' es 65
    ...
 */

int main(){

    Pint A, C, F;
    Pint* B;
    Pchar D, E;
    char G;
    int H;

    H = 66;
    G = 'E';
    A = new int;
    F = new int;
    (*A) = 64;
    B = &F;
    (*F) = (*A);
    cout << (**B) << (*A) << endl;

    D = (Pchar)(*B);
    E = (Pchar)A;
    C = (Pint)D;
    (*C) = H;
    cout << (*D) << (*C) << (**B) << endl;

    (*F) = (*C) + 3;
    if ((*D) == G) {
        cout << (*E) << (*C) << endl;
    }
}
```

```
}

(*E) = 'A';
(*F) = (*C) - (*A);
if (F == (*B))
    cout << (*A) << (*C) << endl;

delete A;
delete F;

return 0;
}

/*
Resultado:
        64 64
        B 66 66
        @ 69
        65 4
*/
```

Ejercicio 10: Hacer el seguimiento

```
/* Punteros
 * Ejercicio Tipo Parcial N 7
 */

#include <stdlib.h>
#include <iostream>

using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
Datos:
    '@' es 64
    'A' es 65
    ...
*/

int main(){

    Pint A, C, F;
    Pint* B;
    Pchar D, E;
    char G;
```

```

int H;

H = 67;
G = 'A';
A = new int;
(*A) = 64;
B = &C;
F = A;
C = &H;
cout << (**B) << (*A) << endl;

D = (Pchar)(*B);
E = (Pchar)A;
cout << (*D) << (*C) << (*E) << endl;

(*D) = G;
if ((*C) == 67) {
    cout << (*E) << (*C) << endl;
}

(*A) = 66;
while ((*A) > 0) {
    cout << (*A) << (*E) << endl;
    (*F) = (*F) - H;
    (*A) = (*A) - 1;
}

delete A;

return 0;
}

/*
Resultado:
67 64
C 67 @
66 B
*/

```

Ejercicio 11: hacer el seguimiento

```

/* Punteros
 * Ejercicio Tipo Parcial N8
 */

#include <stdlib.h>
#include <iostream>

```

```
using namespace std;

typedef int* Pint;
typedef char* Pchar;

/*
    Datos:
    '@' es 64
    'A' es 65
    ...
*/

int main(){

    Pint A, C;
    Pint* B;
    Pint* F;
    Pchar D, E;
    char G;
    int H;

    H = 70;
    G = 'A';
    A = new int;
    (*A) = 65;
    F = new Pint;
    B = &A;
    (*F) = A;
    cout << (**B) << (**F) << endl;

    D = (Pchar)(*B);
    E = (Pchar)&H;
    C = (Pint)D;
    (*C) = H;

    if ((*A) == H ) {
        cout << (**B) << (*E) << (*C) << endl;
    }

    (*D) = G;
    (*A) = (**F) - 1;
    (*E) = (char) (**F);
    cout << (*C) << H;

    delete A;
    delete F;
}
```

```
    return 0;
}

/**
 * Resultados
 * 65 65
 * 70 F 70
 * 64 64
```

Ejercicio 12: hacer el tatei con una matriz de punteros a char, siendo ' ' el casillero vacío, 'X' para el jugador 1 y 'O' para el jugador 2.