

Nombre de la práctica	WHILE			No.	9
Asignatura:	Probabilidad y Estadística	Carrera:	Ingeniería en Sistemas Computacionales	Duración de la práctica (Hrs)	

NOMBRE DEL ALUMNO: Jesús Navarrete Martínez

GRUPO: 3401

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Actividades en aula de clases y en equipo personal

III. Material empleado:

- Laptop
- Visual Studio

Ejercicio 1 Imprimir tres veces la palabra Hola

```

1  #include <stdio.h>
2
3  int main(){
4      int contador = 0;
5      while(contador<3){
6          printf("Hola \n");
7          contador++;
8      }
9      printf("Fin");
10     return 0;
11 }

```

```

Hola
Hola
Hola
Fin

```

Ejercicio 2 Imprimir tres veces la palabra Hola

```
1  #include <stdio.h>
2
3  int main(){
4      int contador = 5;
5      while(contador<3){
6          printf("Hola \n");
7          contador++;
8      }
9      printf("Fin");
10     return 0;
11 }
```



Ejercicio 3 Imprimir los números del 1 al 1000

```
1  #include <stdio.h>
2
3  int main(){
4      int numero = 1;
5
6      while (numero<=1000){
7          printf("%d, ", numero);
8          numero++;
9      }
10     return 0;
11 }
```

PS C:\Users\Vanesa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9 WHILE> .\474

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000,

PS C:\Users\Vanesa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9 WHILE> |

Ejercicio 4

Imprimir los cuadrados y los cubos de los primeros quinientos números naturales.

El cuadrado de un número se define como: x^2 El cubo de un número se define como: x^3

La salida debe ser la siguiente:

1, 1, 1

2, 4, 8

3, 9, 27

4, 16, 64

```
1  #include <stdio.h>
2
3  int main(){
4      int numero = 1;
5
6      while (numero<=500){
7
8          printf("%d,%d,%d ", numero, numero*numero, numero*numero*numero);
9          printf("\n");
10         numero ++;
11     }
12     return 0;
13 }
```



1,1,1
2,4,8
3,9,27
4,16,64
5,25,125
6,36,216
7,49,343
8,64,512
9,81,729
10,100,1000
11,121,1331
12,144,1728
13,169,2197
14,196,2744
15,225,3375
16,256,4096
17,289,4913
18,324,5832
19,361,6859
20,400,8000
21,441,9261
22,484,10648
23,529,12167
24,576,13824
25,625,15625
26,676,17576
27,729,19683
28,784,21952
29,841,24389
30,900,27000
31,961,29791
32,1024,32768
33,1089,35937
34,1156,39304

Ejercicio 5

Imprimir todos los números divisibles entre 3 mayores a 0 y menores a mil.

Ejemplo: 3, 6, 9, 12, 15, 18, 21, 999

```
1  #include <stdio.h>
2
3  int main (){
4      int i = 1;
5
6      printf("Ejercicio 1--->Numeros divisbles entre 3 mayores a 0 y menores a 1000\n");
7
8      while(i <= 1000){
9          if(i % 3 == 0){
10             printf("%d \n", i);
11         }
12         i++;
13     }
14     return 0;
15 }
```

```
3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96, 99, 102, 105, 108, 111, 114, 117, 120, 123, 126, 129, 132, 135, 138, 141, 144, 147, 150, 153, 156, 159, 162, 165, 168, 171, 174, 177, 180, 183, 186, 189, 192, 195, 198, 201, 204, 207, 210, 213, 216, 219, 222, 225, 228, 231, 234, 237, 240, 243, 246, 249, 252, 255, 258, 261, 264, 267, 270, 273, 276, 279, 282, 285, 288, 291, 294, 297, 300, 303, 306, 309, 312, 315, 318, 321, 324, 327, 330, 333, 336, 339, 342, 345, 348, 351, 354, 357, 360, 363, 366, 369, 372, 375, 378, 381, 384, 387, 390, 393, 396, 399, 402, 405, 408, 411, 414, 417, 420, 423, 426, 429, 432, 435, 438, 441, 444, 447, 450, 453, 456, 459, 462, 465, 468, 471, 474, 477, 480, 483, 486, 489, 492, 495, 498, 501, 504, 507, 510, 513, 516, 519, 522, 525, 528, 531, 534, 537, 540, 543, 546, 549, 552, 555, 558, 561, 564, 567, 570, 573, 576, 579, 582, 585, 588, 591, 594, 597, 600, 603, 606, 609, 612, 615, 618, 621, 624, 627, 630, 633, 636, 639, 642, 645, 648, 651, 654, 657, 660, 663, 666, 669, 672, 675, 678, 681, 684, 687, 690, 693, 696, 699, 702, 705, 708, 711, 714, 717, 720, 723, 726, 729, 732, 735, 738, 741, 744, 747, 750, 753, 756, 759, 762, 765, 768, 771, 774, 777, 780, 783, 786, 789, 792, 795, 798, 801, 804, 807, 810, 813, 816, 819, 822, 825, 828, 831, 834, 837, 840, 843, 846, 849, 852, 855, 858, 861, 864, 867, 870, 873, 876, 879, 882, 885, 888, 891, 894, 897, 900, 903, 906, 909, 912, 915, 918, 921, 924, 927, 930, 933, 936, 939, 942, 945, 948, 951, 954, 957, 960, 963, 966, 969, 972, 975, 978, 981, 984, 987, 990, 993, 996, 999,
PS C:\Users\Vanessa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9_WHILE> |
```

Ejercicio 6

Imprimir todos los números que son divisibles entre 2 entre 7, mayores a 0 y menores a mil.

Ejemplo: 14, 28, 42, 56, ...

```
1  #include <stdio.h>
2
3  int main (){
4      int i = 1;
5
6      printf("Ejercicio 2--->Numeros divisbles entre 2 y entre 7 mayores a 0 y menores a 1000\n");
7
8      while(i <= 1000){
9          if(i % 2 == 0){
10             if(i % 7 == 0){
11                 printf("%d \n", i);
12             }
13         }
14         i++;
15     }
16     return 0;
17 }
```



```
PS C:\Users\Vanessa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9_WHILE> gcc ejercicio2.c
PS C:\Users\Vanessa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9_WHILE> ./a
Ejercicio 2---->Numeros divisibles entre 2 y entre 7 mayores a 0 y menores a 1000
14, 28, 42, 56, 70, 84, 98, 112, 126, 140, 154, 168, 182, 196, 210, 224, 238, 252, 266, 280, 294, 308, 322, 336, 350, 364, 378, 392, 406, 420, 434, 448, 462, 476, 490, 504, 518, 532,
546, 560, 574, 588, 602, 616, 630, 644, 658, 672, 686, 700, 714, 728, 742, 756, 770, 784, 798, 812, 826, 840, 854, 868, 882, 896, 910, 924, 938, 952, 966, 980, 994,
PS C:\Users\Vanessa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9_WHILE> |
```

Ejercicio 7

Escribir todos los enteros positivos menores que 100 omitiendo aquellos que son divisibles, por ejemplo

```
1  #include <stdio.h>
2
3  int main(){
4      int i = 1;
5
6      printf("Ejercicio 3 ----> Enteros positivos menores a 100 que no son divisibles entre 7: \n");
7
8      while(i < 100){
9          if(i % 7 != 0) {
10             printf("%d\n", i);
11         }
12         i++;
13     }
14     return 0;
15 }
```

```
Ejercicio 3 ----> Enteros positivos menores a 100 que no son divisibles entre 7:
1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 22, 23, 24, 25, 26, 27, 29, 30, 31, 32, 33, 34, 36, 37, 38, 39, 40, 41, 43, 44, 45, 46, 47, 48, 50, 51, 52, 53, 54, 55,
57, 58, 59, 60, 61, 62, 64, 65, 66, 67, 68, 69, 71, 72, 73, 74, 75, 76, 78, 79, 80, 81, 82, 83, 85, 86, 87, 88, 89, 90, 92, 93, 94, 95, 96, 97, 99,
PS C:\Users\Vanessa Hernandez\Documents\Nueva carpeta\UNIVERSIDAD\SEMESTRE 4\METODOS NUMERICOS\9_WHILE> |
```

Ejemplo 8 Do While

```
1  #include <stdio.h>
2
3  int main (){
4      int i = 0;
5      do
6      {
7          printf("Valor de i = %d \n", i);
8          i++;
9      } while (i<3);
10     return 0;
11 }
```

```
Valor de i = 0  
Valor de i = 1  
Valor de i = 2  
PS C:\Users\Marcelo>
```

Conclusión

En conclusión, la sentencia "while" ofrece una manera versátil de repetir un bloque de código mientras una condición particular siga siendo verdadera. Esta estructura es crucial para crear bucles en situaciones donde el número de repeticiones no es predecible y depende de una condición que puede cambiar..