**//file: pointervar.c**

**#include <stdio.h>**

**int main(void)**

**{**

**int data = 100;**

**int \*ptrint;**

**ptrint = &data;**

**printf("십진수소수값: %u\n", ptrint);**

**printf("%u\n", &data);**

**printf("%u\n", data);**

**printf("%u\n", \*ptrint);**

**puts("");**

**printf("포인터변수의크기: %d\n", sizeof(ptrint));**

**printf("%d\n",sizeof(data));**

**printf("%d\n",sizeof(&data));**

**printf("%d\n",sizeof(\*ptrint));**

**puts("");**

**printf("%d\n", \*ptrint);**

**printf("%d\n", data);**

**printf("%d\n", ptrint);**

**printf("%d\n", &data);**

**return 0;**

**}**

8-1

//file: address.c

#include <stdio.h>

int main (void)

{

int input;

printf("정수입력: ");

scanf("%d", &input);

printf("입력값: %d\n", input);

printf("주소값: %u(십진수), %p(16진수)\n", &input, &input);

printf("주소값: %d(십진수), %p(16진수)\n", &input, &input);

printf("주소값크기: %d\n", sizeof (&input));

return 0;

}

8-2

//file: pointervar.c

#include <stdio.h>

int main (void)

{

int data = 100;

int \*ptrint;

ptrint = &data;

printf("십진수주소값: %u\n", ptrint);

printf("포인터변수의크기: %d\n", sizeof(ptrint));

return 0;

}

8-3

//file: nullpointer.c

#include <stdio.h>

int main(void)

{

int \*ptr1, \*ptr2;

ptr1 = NULL;

printf("%u\n", ptr1);

printf("%u\n", ptr2);

return 0;

}

8-4

//file: dereference.c

#include <stdio.h>

int main (void)

{

int data = 100;

char ch = 'A';

int \*ptrint = &data;

char \*ptrchar = &ch;

printf("간접참조출력: %d %c\n", \*ptrint, \*ptrchar);

\*ptrint = 200;

\*ptrchar = 'B';

printf("간접참조출력: %d %c\n", data, ch);

return 0;

}

8-5

//file: pointerarith.c

#include <stdio.h>

int main(void)

{

char \*pc = (char\*)100;

int \*pi = (int\*)100;

double \*pd = (double\*)100;

printf("%u %u %u\n", pc-1, pc, pc+1);

printf("%u %u %u\n", pi-1, pi, pi+1);

printf("%u %u %u\n",pd-1, pd, pd+1);

return 0;

}

8-6

//file: pointerpriority.c

#include <stdio.h>

int main (void)

{

int a[4] = {1, 3, 6, 8};

int \*pa = &a[0];

printf("%d %d %d %d\n", \*(pa), \*(pa+1), \*(pa+2), \*(pa+3));

printf("%d %d %d %d\n", pa[0], pa[1], pa[2], pa[3]);

puts("");

printf("%d ", ++\*pa);

printf("%d ", \*++pa);

printf("%d ",\*pa++);

printf("%d ", --\*pa++);

printf("%d\n", (\*pa)--);

printf("%d %d %d %d\n", a[0], a[1], a[2], a[3]);

return 0;

}

8-7

//file: ptypecast.c

#include <stdio.h>

int main (void)

{

char c[4] = {'A', '\0', '\0', '\0'};

int \*pi = (int \*)&c[0];

printf("%d %c\n", (int)c[0], c[0]);

printf("%d %c\n", \*pi, (char)\*pi);

return 0;

}

\*\*참조

#include <stdio.h>

void main()

{

int arr[5];

int i;

int sum;

int max, min;

printf("평균나이프로그램\n\n");

printf("나이를입력하세요\n");

for (i=0; i<5; i++)

scanf("%d", &arr[i]);

printf("\n");

sum = 0;

for (i=0; i<5; i++)

sum = sum + arr[i];

printf("작성한나이의평균은[%d]세입니다.\n\n", sum/5);

max = 0;

for (i=0; i<5; i++)

{

if (max < arr[i])

{

max = arr[i];

}

}

printf("이중최연장자는[%d]세입니다.\n\n", max);

min = 0;

for (i=0; i<5; i++)

{

if (min > arr[i])

{

min = arr[i];

}

}

printf("이중최연소자는[%d]세입니다.\n\n", min);

}

8-12

//file: tdarypointer1.c

#include <stdio.h>

#define ROW 2

#define COL 3

int main (void)

{

int td[][COL] = {{8, 5, 4}, {2, 7, 6}};

int i = 0, j = 0, cnt = 0;

printf("%d, %d, %d\n", sizeof(td), sizeof(td[0]), sizeof(td[1]));

printf("%u, %u %u\n", td, td[0], td[1]);

printf("%u %u \n", &td[0][0], &td[1][0]);

\*\*td = 10;

\*td[1] = 20;

for (i = 0; i < ROW; i++)

{

for (j = 0; j < COL; j++, cnt++)

{

printf("%d, %d ", \*(\*td + cnt), \*(td[i] + j));

}

puts("");

}

return 0;

}

8-13

//file: arrayparam.c

#include <stdio.h>

int sumary(int \*ary, int SIZE);

int main (void)

{

int i, sum = 0;

int point[] ={95, 88, 76, 54, 85, 33, 65, 78, 99, 82};

int \*address = point;

int aryLength = sizeof (point) / sizeof (int);

for (i = 0; i < aryLength; i++)

sum += \*(point+i);

printf("메인에서구한합은%d\n", sum);

address = point;

printf("함수sumary() 호출로구한합은%d\n", sumary(point, aryLength));

printf("함수sumary() 호출로구한합은%d\n", sumary(&point[0], aryLength));

printf("함수sumary() 호출로구한합은%d\n", sumary(address, aryLength));

return 0;

}

int sumary( int \*ary, int SIZE)

{

int sum = 0, i = 0;

for (i = 0; i < SIZE; i++)

{

sum += \*(ary++);

}

return sum;

}