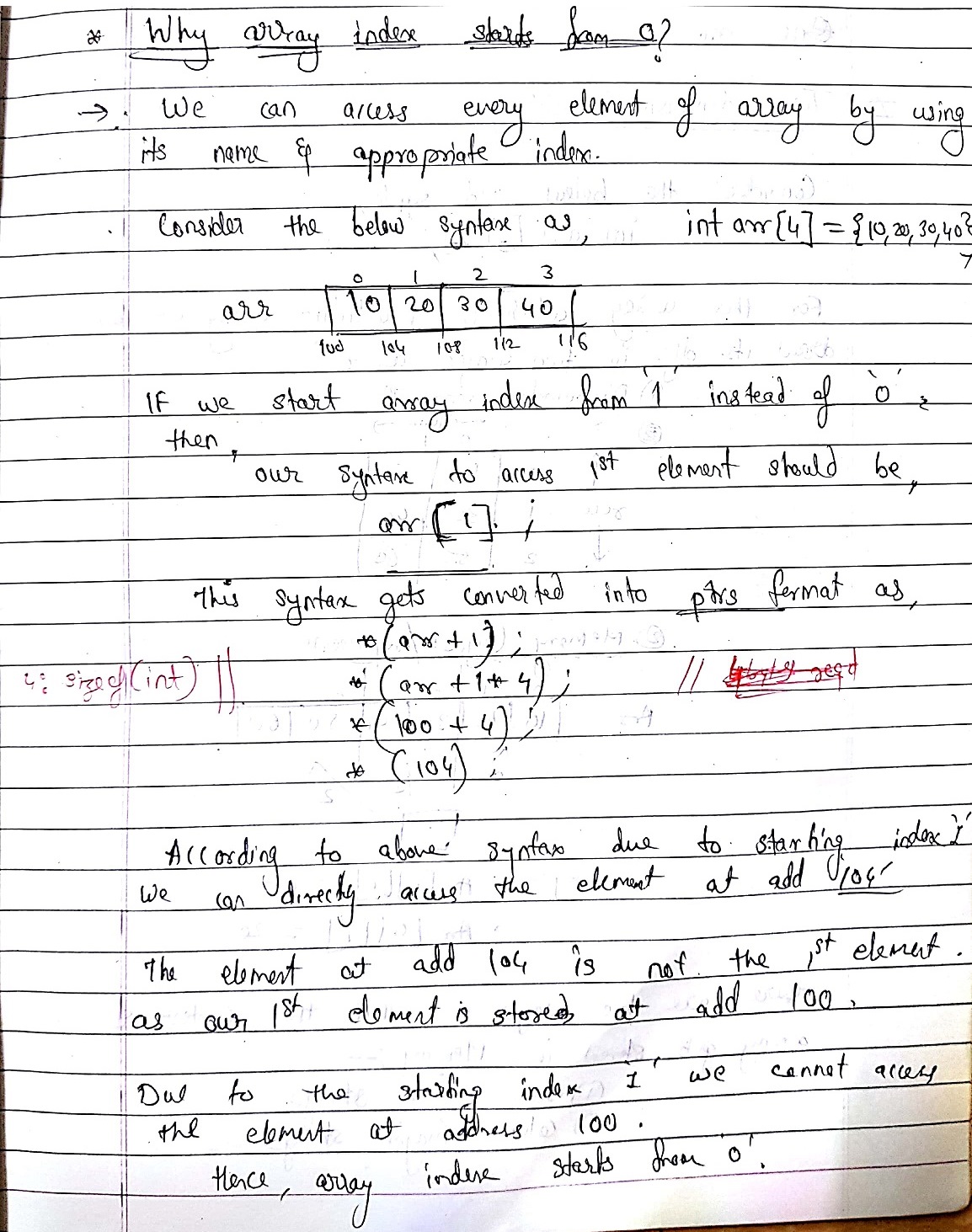
///////////////////////////////////////////////////////////////////////////

//Why Array index starts from 0 :-



//int Arr[5];

Arr++; //Error

Note:- Arr++ can never be used as we are trying to modify the base address itself.

////////////////////////////////////////////////////////////////////////////

//One-Dimensional Array

#include<stdio.h>

#include<stdlib.h>

int main()

{

int \*p=NULL;

int element=0,i=0;

printf("Enter total no. of the elemets\n");

scanf("%d" , &element);

//memory allocation

p = (int \*)malloc(sizeof(int) \* element);

//if memory is not allocated

If(p==NULL)

{

Printf(“Unable to allocate memory”);

Return -1;

}

//accept elements from user

Printf(“Enter elements”);

for(i = 0 ; I < elements ; i++)

{

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

}

//print the entered elements

printf("Elemets in 1D array are\n");

for(i = 0 ; I < elements ; i++)

{

printf(" %d " , p[i]);

}

printf("Free the allocated memory for the 1D array\n");

free(p);

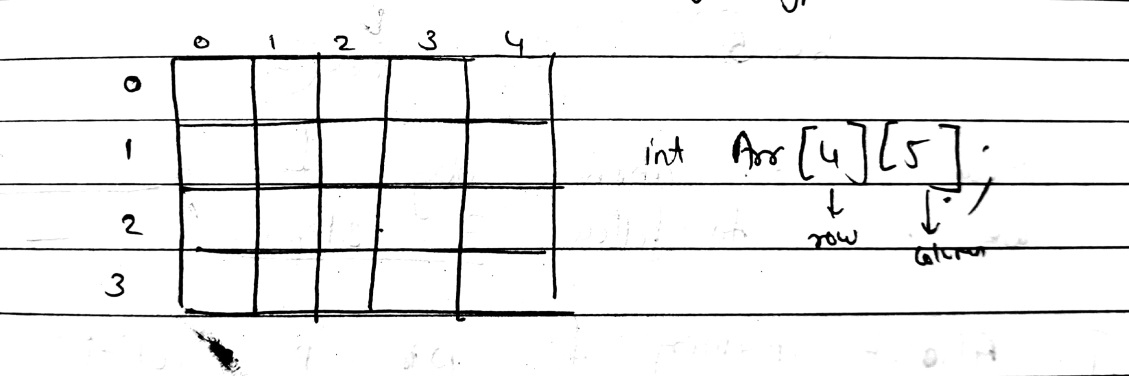
return 0;

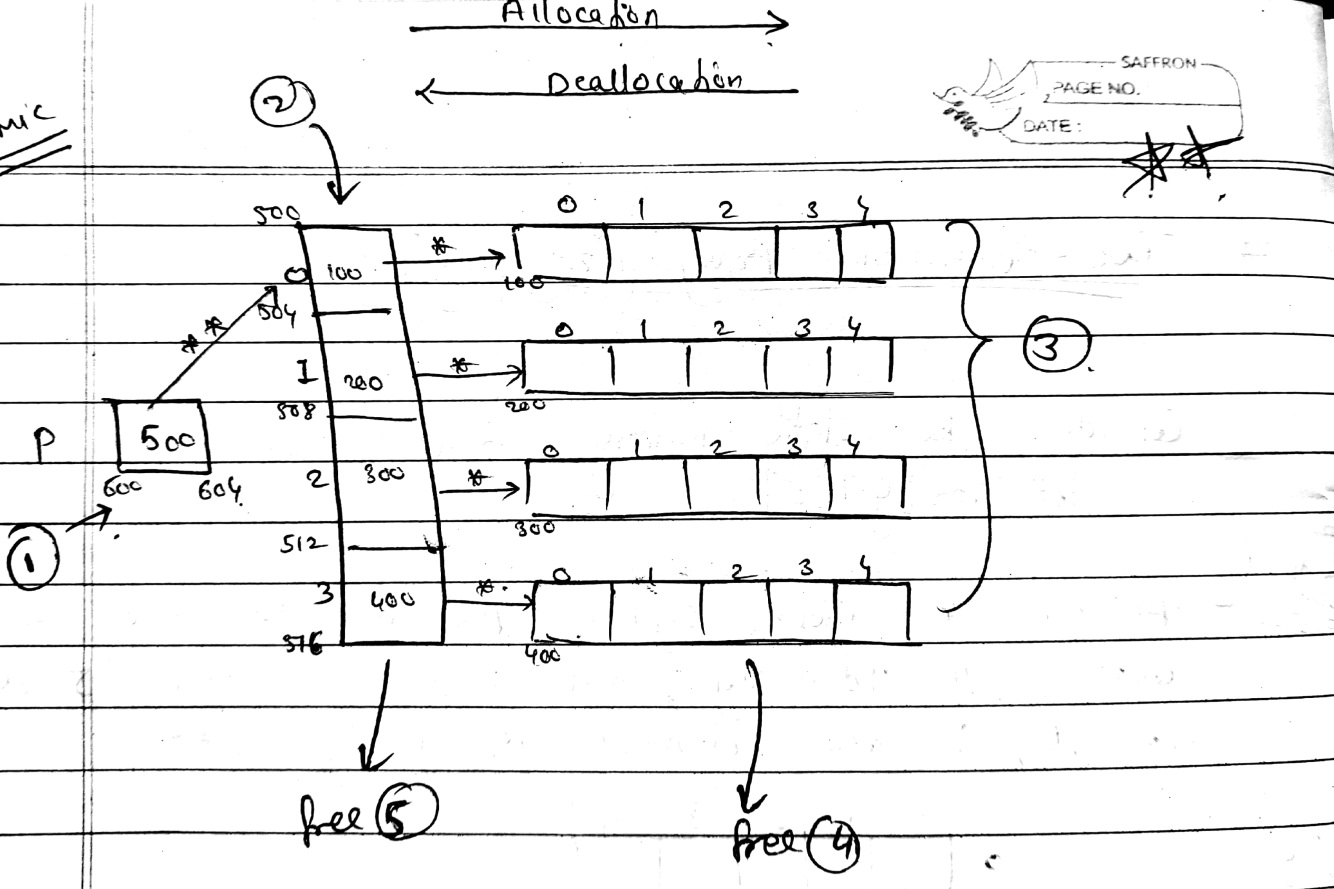
}

////////////////////////////////////////////////////////////////

////////////////////////////////////////////////////////////////////////////

//Two-Dimensional Array





int main()

{

int col=0,row=0,i=0,j=0;

int \*\*p = NULL; //step I

printf("Enter no. of rows and columns\n");

scanf("%d %d" , &row, &col);

//Step II

p = (int\*\*)malloc(row\* sizeof(int));

//STEP III

for(i = 0 ; I < row ; i++)

{

p[i] = (int \*)malloc(col \* sizeof(int));

}

printf("Enter elemets \n");

for(i = 0 ; I < row ; i++)

{

for(j = 0 ; j < col ; j++)

{

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

}

}

printf("Your entered elemets are\n");

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

{

printf("\n");

for(j = 0 ; j < col ; j++)

{

printf(" %d " , p[i][j]);

}

}

//Step IV

Free the allocated memory

for(i = 0 ; I < row ; i++)

{

free(p[i]);

}

//step V

Free(p);

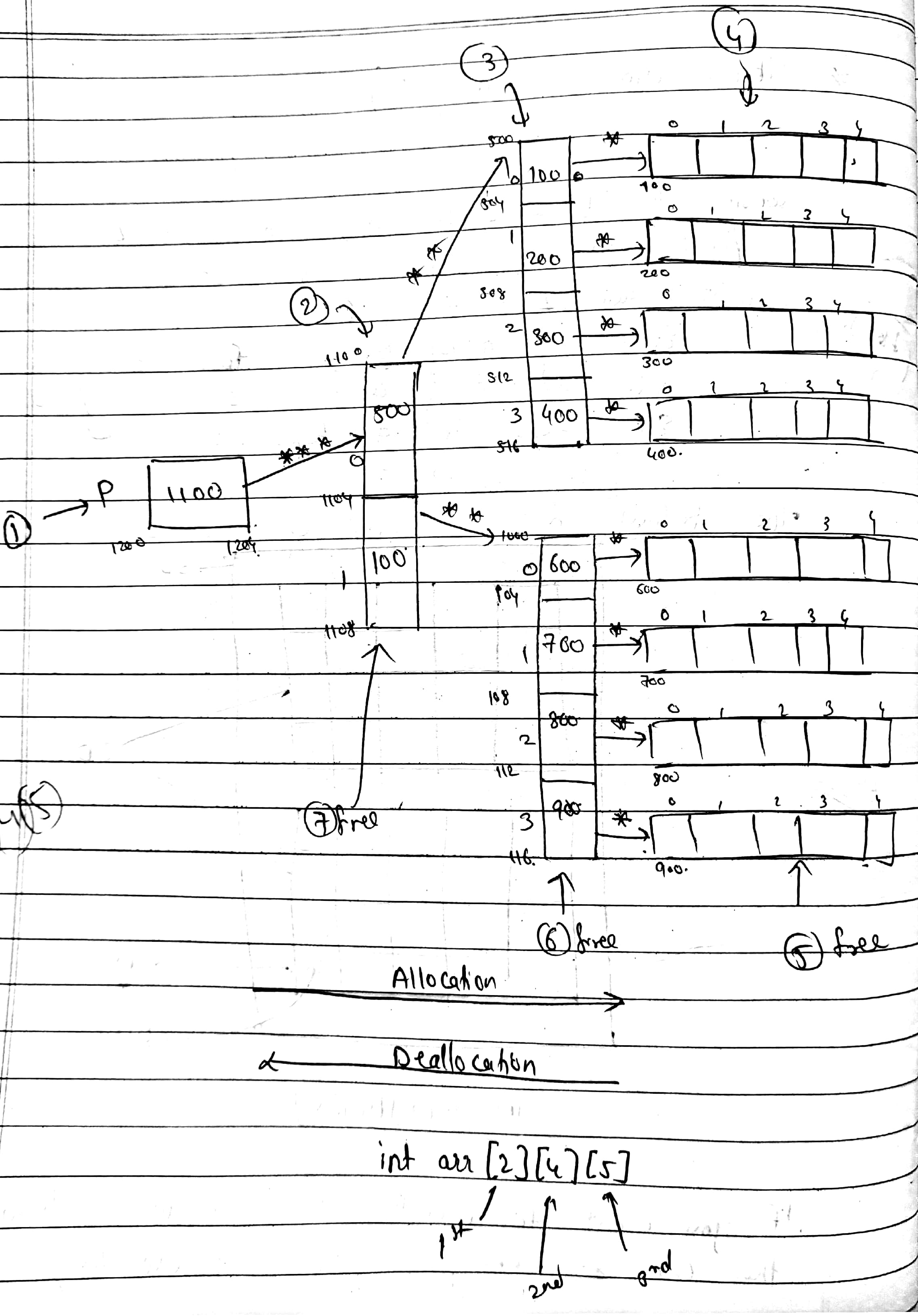
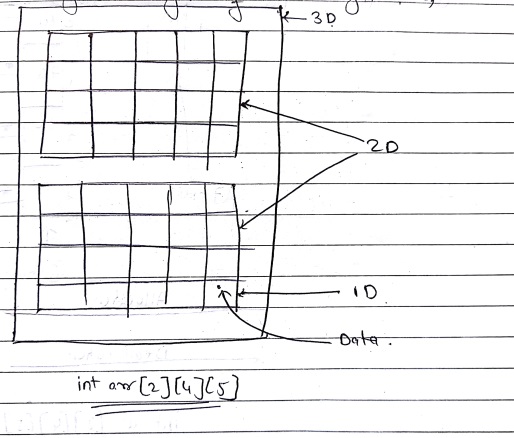
return 0;

}

///////////////////////////////////////////////////////////////////////////////////////////////////////////

////////////////////////////////////////////////////////////////////////////

//Three Dimensional Array



//Memory Allocation steps:- 1,2,3,4

// Memory Deallocation steps:- 5,6,7

int main()

{

int first=0 , second=0 , third=0 , I=0 , j=0 , k=0;

//step I

int \*\*\*p=NULL;

printf("Enter dimentions\n");

scanf("%d%d%d " , &first,&second,&third);

//Step II

p=(int \*\*\*)malloc(first \* sizeof(int\*\*));

//Step III

for(i = 0 ; I < first ; i++)

{

p[i] = (int \*\*)malloc(second \*sizeof(int\*));

}

//Step IV

for(i = 0 ; I < first ; i++)

{

for( j = 0; j < second ; j++)

{

p[i][j] = (int \*)malloc(third \* sizeof(int));

}

}

printf("Enter elements\n ");

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

{

for(j =0 ; j < second ; j++)

{

for(k = 0 ; k < third ; k++)

{

scanf("%d" , &p[i][j][k]);

}

}

}

//Deallocation starts, Step V

for(i = 0 ; I < first ; i++)

{

for(j = 0 ; j < second ; j++)

{

Free(p[i][j]);

}

}

//Step VI

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

{

Free(p[i]);

}

//Step VII

Free(p);

return 0;

}

//4d tracing :-

