|  |  |  |
| --- | --- | --- |
| 1 | int A[ 10]={1,2,3,4,5,6,7,8,9,10};  print (A, sizeof(A)/sizeof(A[0]) ); | void print ( int ar[], size\_t n );  //void print ( int \*p , size\_t n );  void print ( int(& ar)[10] ); |
| 2 | int A[N][M]={1,2,3,4,5,6,7,8,9,10};  print (A, sizeof(A)/sizeof(A[0])); | const size\_t N=4, M=3;  void print ( int ar [][M], size\_t n );  //void print ( int (\*p)[M] , size\_t n ); |
| 3 | const char A[]={“We don\’t need no education”};  print (A); | void print ( const char ar [] );  //void print ( const char \* p ); |
| 4 | const char\* A[]=  {“We”, “don\’t”, “need”, “no”, “education”};  print (A, sizeof(A)/sizeof(A[0])); | void print ( const char \* ar [] , size\_t n );  //void print ( const char \* \* p , size\_t n ); |
| 5 | int\* A[N]; size\_t m=< … >;  for ( int i=0; i<N;i++)  {  A[i] =new int[M];  for (int j=0; j<m;j++)  A[i][j] =< … >;  }  print (A, N, M); | const size\_t N=4;  void print ( int \* ar [], size\_t n, size\_t m);  //void print ( int \*\* ar , size\_t n, size\_t m); |
| 6 | int\* A; size\_t n =< … >;  A =new int[n];  for (int i=0; i < n; i++)  A[i] =< … >;  print (A, n); | //как в 1)  void print ( int \*p , size\_t n );  //void print ( int ar [], size\_t n ); |
| 7 | int\* p ;  size\_t n=< … >, m=< … >;  p= new int [n\*m];  int\*\*pp=new int\* [n];  for ( int i=0; i<n; i++)  {  pp [i] =p + i\*m;  for (int j=0; j<m;j++) pp[i][j] =< … >;  }  print (pp, n, m); | //как в 5)  void print ( int \*\*p , size\_t n, size\_t m); |
| 8 | int\*\*pp ;  size\_t n=< … >, m=< … >;  pp= new int\* [n];  for ( int i=0; i<n;i++)  {  pp[i] =new int[n];  for (int j=0; j<m; j++) pp[i][j] =< … >;  }  print (pp, n, m); | //как в 5)  void print ( int \*\*p , size\_t n, size\_t m); |