emacs@behrooz-kntu-PC File Edit Options Buffers Tools C Help Save Undo V \* convolve.c /\* Standard includes \*/ #include cassert.h>
#inclu SSH /\* Our includes \*/ #include "base.h" #include "error.h" Session 17 #include "klt util.h" /\* printing \*/ #define MAX KERNEL WIDTH 71 typedef street
int widt;
float data[MAX\_KERNEL\_WIDTH]; ConvolutionKernel; /\* Kernels \*/

## Remember: count the ratings

```
int n, r, r1, r2, r3, r4;
r1 = r2 = r3 = r4 = 0:
n = 0;
while (1) {
  printf("Enter rating: ");
                                       1: Bad
  scanf("%d", &r);
 if (r==1)
                                       2: Average
   r1++:
  else if (r== 2)
   r2++;
  else if (r== 3)
```

r3++:

r4++;

break;

continue:

else {

n++;

else if (r== 4)

else if (r==-1)

puts("invalid number!");

// write your code here



```
case 1:
 r1++:
 break:
case 2:
 r2++:
 break:
case 3:
 r3++;
 break:
case 4:
 г4++:
  break;
case -1:
  finish = 1;
  break;
default:
 continue;
  break:
if (finish)
 break:
```

finish = 0; while (1) {

n++;

printf("Enter rating: ");

scanf("%d", &r);

switch (r) {

# What if there are 20 numbers?

```
int n, r, r1, r2, r3, r4;
\Gamma 1 = \Gamma 2 = \Gamma 3 = \Gamma 4 = 0;
n = 0;
while (1) {
  printf("Enter rating: ");
  scanf("%d", &r);
                                                   2:
   r1++:
```

20:

```
if (r== 1)
else if (r== 2)
  r2++;
```





```
else if (r== 3)
 r3++:
else if (r== 4)
 r4++;
else if (r==-1)
 break;
```

else {

n++;

continue:

puts("invalid number!");

// write your code here



```
break;
case -1:
  finish = 1;
  break;
default:
  continue;
  break:
```

printf("Enter rating: ");

scanf("%d", &r);

switch (r) { case 1:

> r1++: break:

case 2:

case 3: r3++;

case 4:

г4++:

if (finish) break;

N++;

r2++;

break:

break:

finish = 0; while (1) {

## What if there are 100 numbers?

finish = 0; while (1) {

printf("Enter rating: ");

scanf("%d", &r);

switch (r) { case 1:

> r1++: break:

case 2:

case 3: r3++;

case 4:

г4++:

case -1:

default:

break;

break;

break:

if (finish) break;

N++;

finish = 1;

continue;

r2++;

break:

break:

```
int n, r, r1, r2, r3, r4;
\Gamma 1 = \Gamma 2 = \Gamma 3 = \Gamma 4 = 0;
n = 0;
while (1) {
  printf("Enter rating: ");
  scanf("%d", &r);
                                                  2:
 if (r== 1)
   r1++:
  else if (r== 2)
```

r2++;

r3++:

r4++;

break;

continue:

else {

n++;

else if (r== 3)

else if (r== 4)

else if (r==-1)

puts("invalid number!");

// write your code here

3:

100:

## What if there are a million numbers?

finish = 0;while (1) {

printf("Enter rating: ");

scanf("%d", &r);

switch (r) { case 1:

> r1++: break:

case 2:

case 3: r3++:

case 4:

r2++:

break:

break:

г4++:

break;

break;

break:

if (finish) break;

n++;

continue;

finish = 1:

case -1:

default:

```
int n, r, r1, r2, r3, r4;
r1 = r2 = r3 = r4 = 0:
n = 0;
while (1) {
  printf("Enter rating: ");
  scanf("%d", &r);
                                            2:
 if (\Gamma == 1)
   r1++:
  else if (r== 2)
                                            3:
   r2++;
  else if (r== 3)
   r3++:
```

1000000:

else if (r== 4)

else if (r==-1)

puts("invalid number!");

// write your code here

r4++;

break;

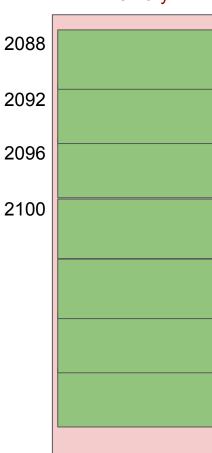
continue:

else {

n++;

# C Arrays

int a[7];



# C Arrays

int a[7];

a[1] = 123;

2088

2092

2096

2100

# C Arrays

```
int a[7];
```

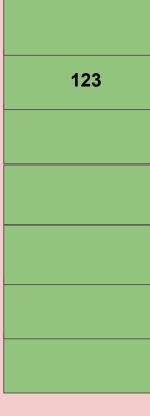
$$a[1] = 123;$$

2088

2092

2100

2096



## C Arrays

subscript

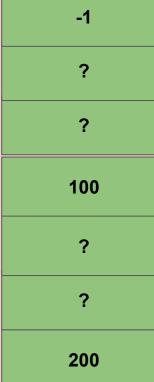
```
int a[7];
a[1] = 123;
index,
```



## C Arrays

```
int a[7];
a[0] = -1;
a[3] = 100;
a[6] = 200;
```

2088	
2002	H



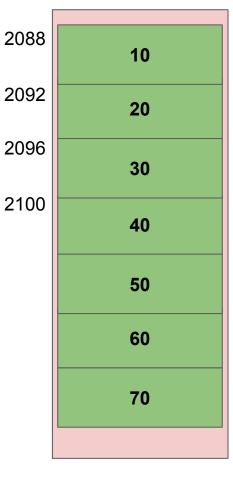
## C Arrays

```
int a[7];
a[0] = -1;
a[3] = 100;
a[6] = 200;
```

2088	-1
2092	?
2096	?
2100	100
	?
	?
	200

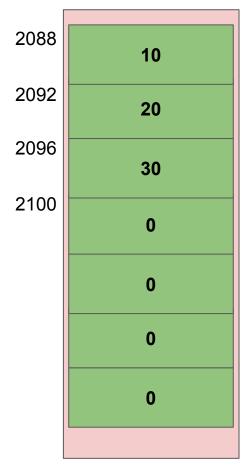
## Initialize Arrays

```
int a[7] = \{10,20,30,40,50,60,70\};
```



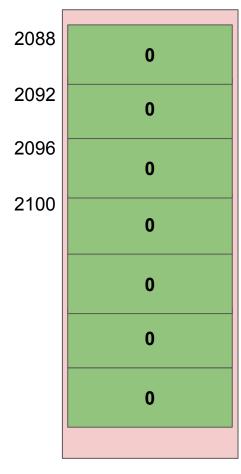
## Initialize Arrays

```
int a[7] = \{10,20,30\};
```



## Initialize Arrays

```
int a[7] = \{0\};
```



### So what?

```
int a[7];
a[0] = -1;
a[3] = 100;
a[6] = 200;
```

2088	-1
2092	?
2096	?
2100	100
	?
	?
	200

2092

2096

2100

### difference with ordinary variables

```
int a[7];
a[0] = -1;
a[3] = 100;
a[6] = 200;
int i = 3;
a[i] = 200;
i += 2;
a[i] = i*i*i;
```

a[i+1] = 300;

```
200
125
300
```

2092

2096

2100

### difference with ordinary variables

```
int a[7];
a[0] = -1;
a[3] = 100;
a[6] = 200;
int i = 3;
a[i] = 200;
i += 2;
```

a[i] = i\*i\*i;

-1
?
?
200
?
125
200

2092

2096

2100

### difference with ordinary variables

```
int a[7];
for (int i = 0; i < 7; i++) {
    a[i] = i*i;
}</pre>
```

?
?
?
?
?
?
?

2088

2092

2096

2100

### difference with ordinary variables

```
int a[7];
for (int i = 0; i < 7; i++) {
    a[i] = i*i;
}
for (int i = 0; i < 7; i++) {
    printf("%d\n", a[i]);
}</pre>
```

0
1
4
9
16
25
36

array1.c

```
while (1) {
  int g;

printf("Enter grade: ");
  scanf("%d", &g);

if (g < 0 || g > 20)
    break;
}
```

```
int count[20];
while (1) {
  int g;

  printf("Enter grade: ");
  scanf("%d", &g);

  if (g < 0 || g > 20)
     break;
}
```

```
int count[21];
while (1) {
  int q;
  printf("Enter grade: ");
  scanf("%d", &g);
 if (g < 0 | | g > 20)
    break;
```

```
int count[21];
for (int i = 0; i < 21; i++)
  count[i] = 0;
while (1) {
 int q;
  printf("Enter grade: ");
  scanf("%d", &g);
 if (g < 0 | | g > 20)
    break;
```

```
int count[21];
for (int i = 0; i < 21; i++)
  count[i] = 0;
while (1) {
 int a:
  printf("Enter grade: ");
  scanf("%d", &g);
 if (q < 0 | | q > 20)
    break;
  count[g] = count[g] + 1;
```

```
int count[21];
for (int i = 0; i < 21; i++)
 count[i] = 0;
while (1) {
 int g;
  printf("Enter grade: ");
  scanf("%d", &g);
 if (q < 0 | | q > 20)
    break:
  count[g]++;
for (int i = 0; i < 21; i++)
  printf("%2d: %d\n", i, count[i]);
```

```
#include <stdio.h>
int main() {
 int count[21];
 for (int i = 0; i < 21; i++)
   count[i] = 0;
 while (1) {
   int g;
   printf("Enter grade: ");
   scanf("%d", &g);
   if (g < 0 | | g >= 21)
     break:
   count[g]++;
 for (int i = 0; i < 21; i++)
   printf("%2d: %d\n", i, count[i]);
 return 0;
```

Change the code so that grades are between 0 and 100.

```
#include <stdio.h>
int main() {
 int count[21];
 for (int i = 0; i < 21; i++)
   count[i] = 0;
 while (1) {
   int g;
   printf("Enter grade: ");
   scanf("%d", &g);
   if (g < 0 | g >= 21)
     break;
   count[g]++;
 for (int i = 0; i < 21; i++)
   printf("%2d: %d\n", i, count[i]);
 return 0;
```

Change the code so that grades are between 0 and 100.

```
#include <stdio.h>
#include <stdio.h>
                                                      #define N 21
                                                      int main() {
int main() {
                                                        int count[N];
 int count[21];
                                                        for (int i = 0; i < N; i++)
  for (int i = 0; i < 21; i++)
                                                          count[i] = 0;
   count[i] = 0;
 while (1) {
                                                        while (1) {
                                                          int g;
   int g;
                                                          printf("Enter grade: ");
   printf("Enter grade: ");
                                                          scanf("%d", &g);
    scanf("%d", &g);
                                                          if (g < 0 || g >= N)
   if (g < 0 | | g >= 21)
                                                            break;
     break:
                                                          count[g]++;
   count[g]++;
                                                        for (int i = 0; i < N; i++)
  for (int i = 0; i < 21; i++)
                                                          printf("%2d: %d\n", i, count[i]);
   printf("%2d: %d\n", i, count[i]);
                                                        return 0;
  return 0;
                                                                                            countarray4.c
```

```
#include <stdio.h>
#define N 21
int main() {
 int count[N];
  for (int i = 0; i < N; i++)
   count[i] = 0;
  while (1) {
   int g;
    printf("Enter grade: ");
    scanf("%d", &g);
   if (g < 0 | | g >= N)
     break;
    count[g]++;
  for (int i = 0; i < N; i++)
    printf("%2d: %d\n", i, count[i]);
  return 0;
                      countarray4.c
```

Is N a variable?

```
#include <stdio.h>
#define N 21
int main() {
 int count[N];
  for (int i = 0; i < N; i++)
   count[i] = 0;
  while (1) {
   int g;
    printf("Enter grade: ");
    scanf("%d", &g);
    if (g < 0 || g >= N)
      break:
    count[g]++;
  for (int i = 0; i < N; i++)
    printf("%2d: %d\n", i, count[i]);
  return 0;
                      countarray4.c
```

```
Is N a variable?
Let's look at the preprocessor output:
$ gcc -E countarray4.c
```

```
int count[21];
for (int i = 0; i < N; i++)
   count[i] = 0;</pre>
```

```
int count[21];
for (int i = 0; i < N; i++)
  count[i] = 0;</pre>
```

```
int count[21];
for (int i = 0; i < N; i++)
   count[i] = 0;</pre>
```

```
int count[21] = {0};
```

```
int count[21];
for (int i = 0; i < N; i++)
  count[i] = 0;</pre>
```

```
int count[21] = {0};
```

```
#define N 21
int main() {
  int count[N] = {0};
```

```
int count[21];
for (int i = 0; i < N; i++)
   count[i] = 0;</pre>
```

```
int count[21] = {0};
```

```
#define N 21
int main() {
  int count[N] = {0};
```

Only works for initializing to zero!

```
#include <stdio.h>
#define N 21
int main() {
 int count[N];
  for (int i = 0; i < N; i++)
   count[i] = 0;
  while (1) {
   int g;
    printf("Enter grade: ");
    scanf("%d", &g);
   if (g < 0 | | g >= N)
      break:
    count[g]++;
  for (int i = 0; i < N; i++)
    printf("%2d: %d\n", i, count[i]);
  return 0;
                       countarray4.c
```

#### Draw a Histogram instead.

```
1:
 2:
 3:
 4:
 5:
 6:
 7:
 8:
 9:
10:
```

```
#include <stdio.h>
                                           #include <stdio.h>
#define N 21
                                           #define N 21
                                           int main() {
int main() {
                                             int count[N] = {0};
 int count[N];
                                                                                        1:
                                             while (1) {
  for (int i = 0; i < N; i++)
                                                                                        2:
                                               int q;
    count[i] = 0;
                                                                                        3:
                                               printf("Enter grade: ");
                                                                                        4:
                                               scanf("%d", &g);
  while (1) {
                                                                                        5:
   int q:
                                               if (q < 0 | | q >= N)
                                                 break:
                                                                                        6:
    printf("Enter grade: ");
                                                                                        7:
    scanf("%d", &g);
                                               count[g]++;
                                                                                        8:
                                             }
    if (q < 0 | | q >= N)
                                                                                        9:
      break:
                                             for (int i = 0; i < N; i++) {
                                                                                      10:
                                               printf("%2d: ", i);
    count[g]++;
                                               for (int j = 0; j < count[i]; j++)
                                                 printf("*");
  for (int i = 0; i < N; i++)
                                               printf("\n");
    printf("%2d: %d\n", i, count[i]);
  return 0:
                                             return 0;
                                                                        countarrayhist.c
                       countarray4.c
```

```
#define N 21
                                                            #define N 21
int main() {
                                                            int main() {
 int count[N] = \{0\};
                                                              int count[N] = {0};
                                                              while (1) {
  while (1) {
   int q;
                                                                int q:
    printf("Enter grade: ");
                                                                printf("Enter grade: ");
                                                                scanf("%d", &g);
    scanf("%d", &g);
                                                                if (q < 0 || q >= N)
    if (q < 0 | | q >= N)
                                                                  break:
      break:
                                                                count[g]++;
    count[g]++;
                                                              for (int i = 0; i < N; i++) {
  for (int i = 0; i < N; i++) {
                                                                printf("%2d: ", i);
    printf("%2d: ", i);
                                                                for (int j = 0; j < count[i]; j++)
    for (int j = 0; j < count[i]; j++)
                                                                  putchar('*');
      printf("*");
                                                                putchar('\n');
    printf("\n");
                                                              return 0;
  return 0;
                              countarrayhist.c
                                                                                        countarrayhist2.c
```

#include <stdio.h>

#include <stdio.h>

### Roll a dice N times!

```
#include <stdio.h>
#include <stdlib.h>
#define SIZE 6
int main() {
  int n;
  int count[SIZE+1];
  for (int i = 1; i <= SIZE; i++)
    count[i] = 0;
  printf("Enter n: ");
  scanf("%d", &n);
  for (int i = 0; i < n; i++) {
   int g = rand()%SIZE + 1;
    count[g]++;
  for (int i = 1; i <= SIZE; i++)
    printf("%2d: %.1f%%\n", i, 100*count[i]/ (double)n);
  return 0;
```