Previous Lecture:
- Simulation
- Application of 1-array and random numbers

Today’s Lecture:
- while loop
- Some array algorithms
- Selection sort

Assigned reading:
- Sec 8.4-8.6

Patterns for doing something \( n \) times

\[
\text{for (int } i=0; i < n; i++)\
{\
// do something
}\]

\[
\text{int } i=0;\
\text{while ( } i < n )\
{\
// do something\
// increment counter\
i++;
}\]

The **while** loop

Syntax:

\[
\text{while ( condition )}\
\text{statement ;}
\]

Pattern for doing something \( n \) times

\[
\text{int } i=0;\
\text{while ( } i < n )\
{\
// do something\
// increment counter\
i++;
}\]

while versus for

Generally...
- Use **for** loop for definite iteration
- Use **while** loop for indefinite iteration

\[
\text{while ( not stopping condition )}\{\
// do something
\}
\]

Some important array algorithms

- Basic operations done on a 1-d array that a programmer should know:
  - Processing an array
  - Searching for the first occurrence of a value
    - Linear search (any array)
    - Binary search (a sorted array)
  - Sorting an array
    - Selection sort
    - Insertion sort

Processing an array

\[
\begin{array}{cccc}
0 & i & n-1 & n \\
\hline
\text{processed} & & & \\
\text{unprocessed} & & & \\
\end{array}
\]

\( i \): position where "process" should be applied next (\( a[0..i-1] \) is processed; \( a[i..] \) is not yet processed)

Read Sec 8.5!!!
Find min value in an array
- General question: Find the location of the min value in array a.
- Suppose you have variable int indexMin; //location of min value
- Then the min value is a[indexMin]

Linear Search

Find location of value v in array a
- Array a is sorted (e.g., in ascending order)
- What if v is not in a?
  Return the location where v should be inserted in a
- Examples:
  - Find 6 in {2,4,4,6,9}: return 3
  - Find 7 in {2,4,4,6,9}: return 4

Binary search

Sorting
- Arrange elements in a list in some order
- Must specify which order
- Sort "in-place"
- Many algorithms:
  - Select sort
  - Insertion sort
  - Bubble sort, ...

Selection Sort
Write a static method selectSort that
- Has a 1-d numeric array as a parameter
- Sorts the numbers in non-descending order in-place

/** Sort array in non-descending order */
public static void selectSort(double[] array){

}