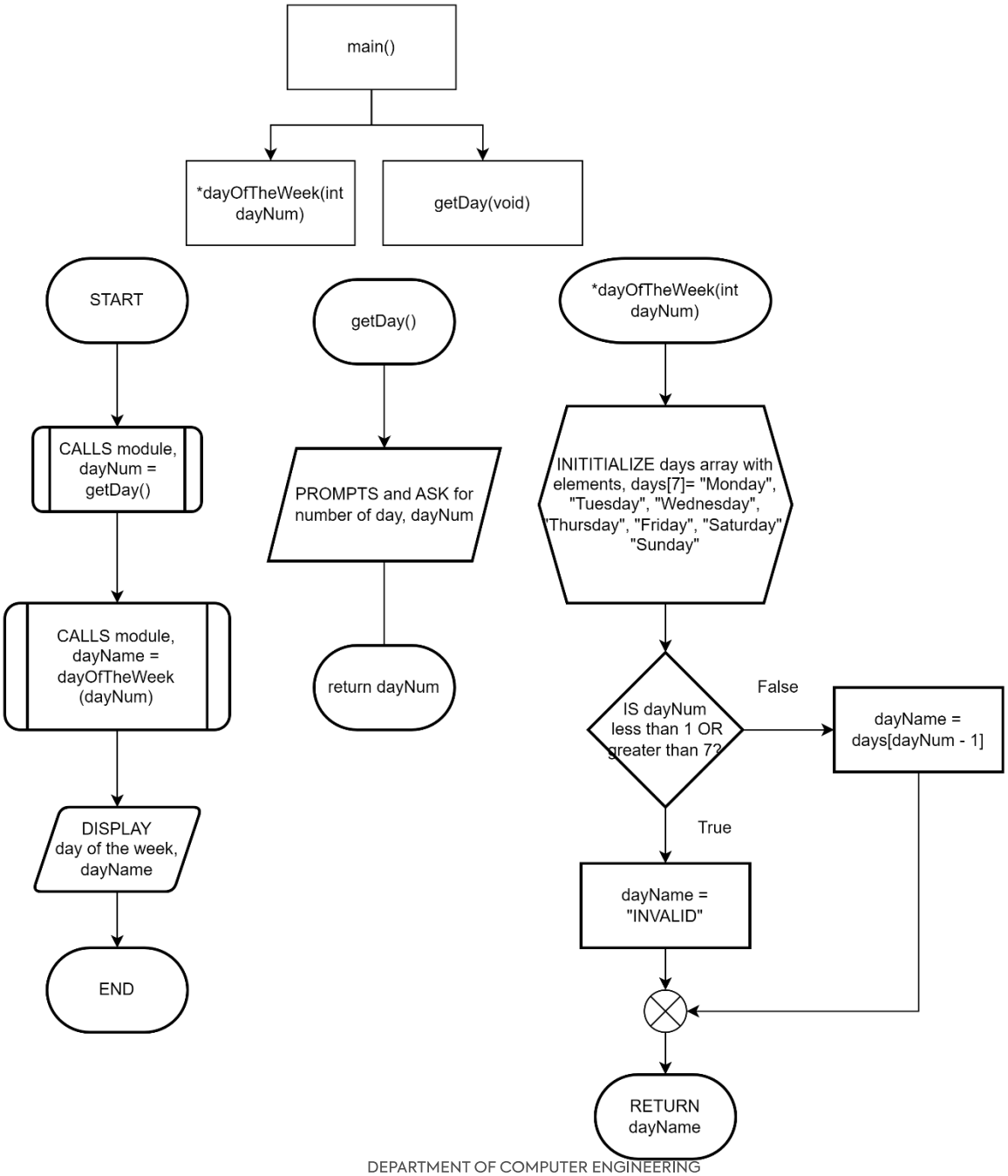
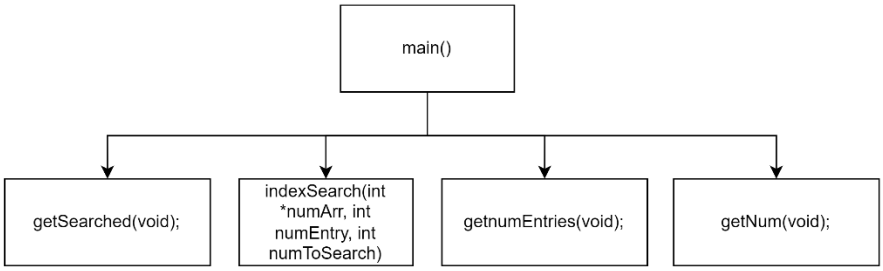
NAME: CHRYS SEAN T. SEVILLA **DATE:** December 1, 2023

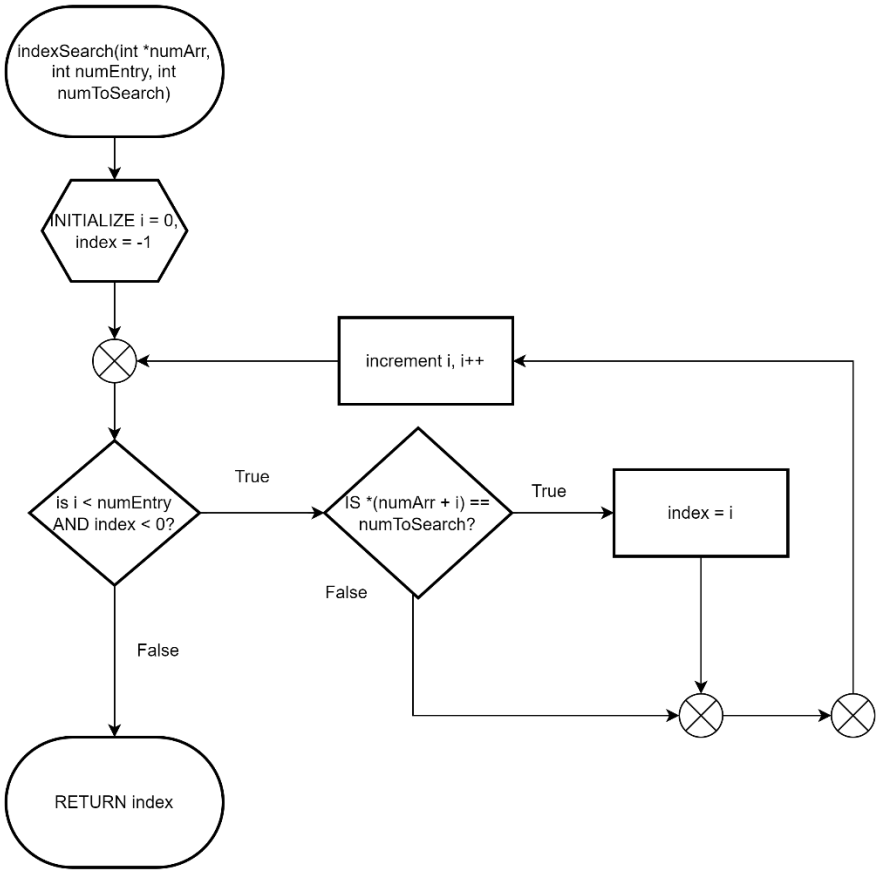
AlgoExercise 8.1

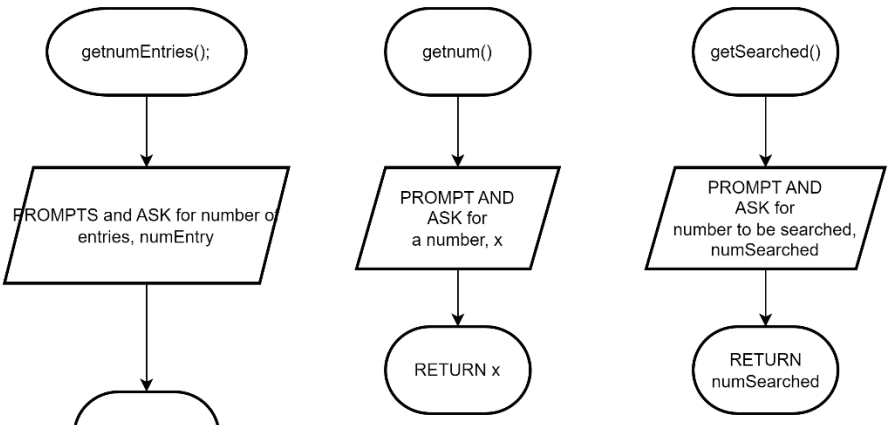
1. Days of the Week.
   A function that sets up an array called days, which contains pointers to the names of the days of the week and return the name of the day from the given day. The function accepts the given day.

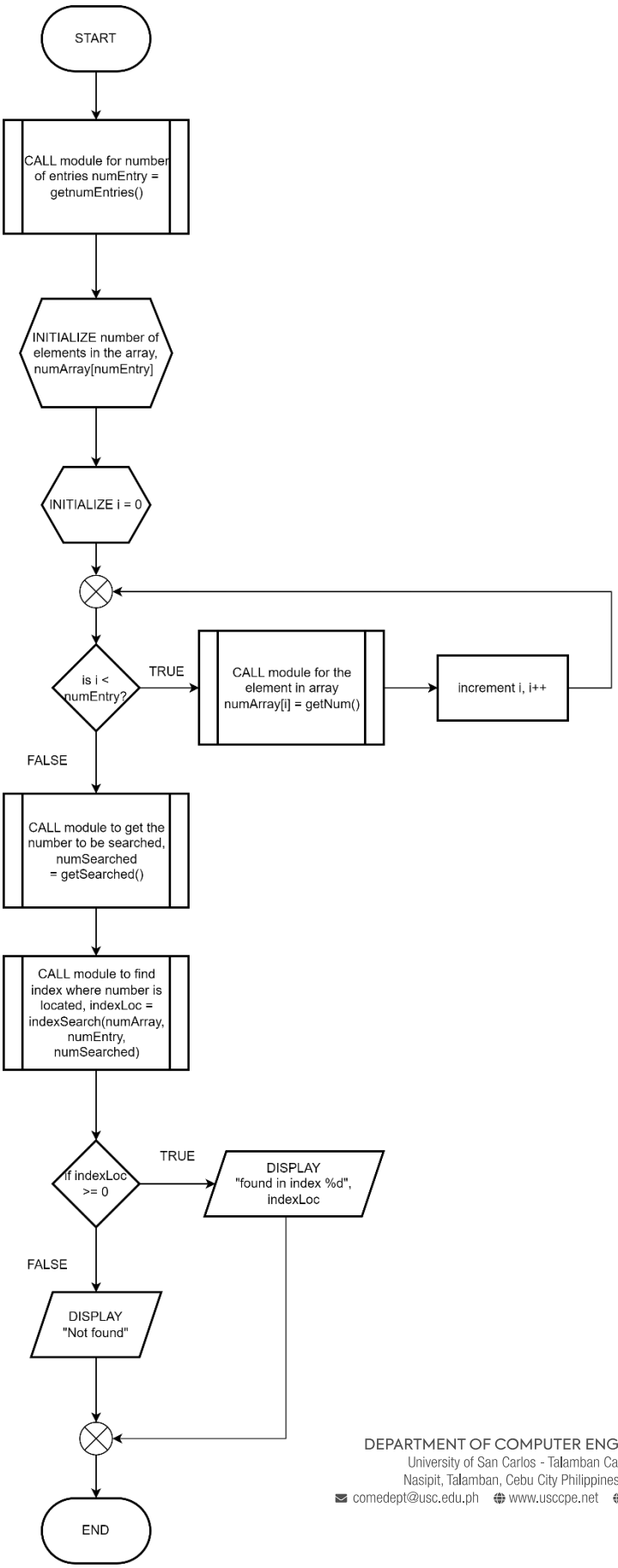


2. Search Element.
   A function to search an element in array using pointers and return the index location. The function accepts the starting address of the array, number of entries and number to search.

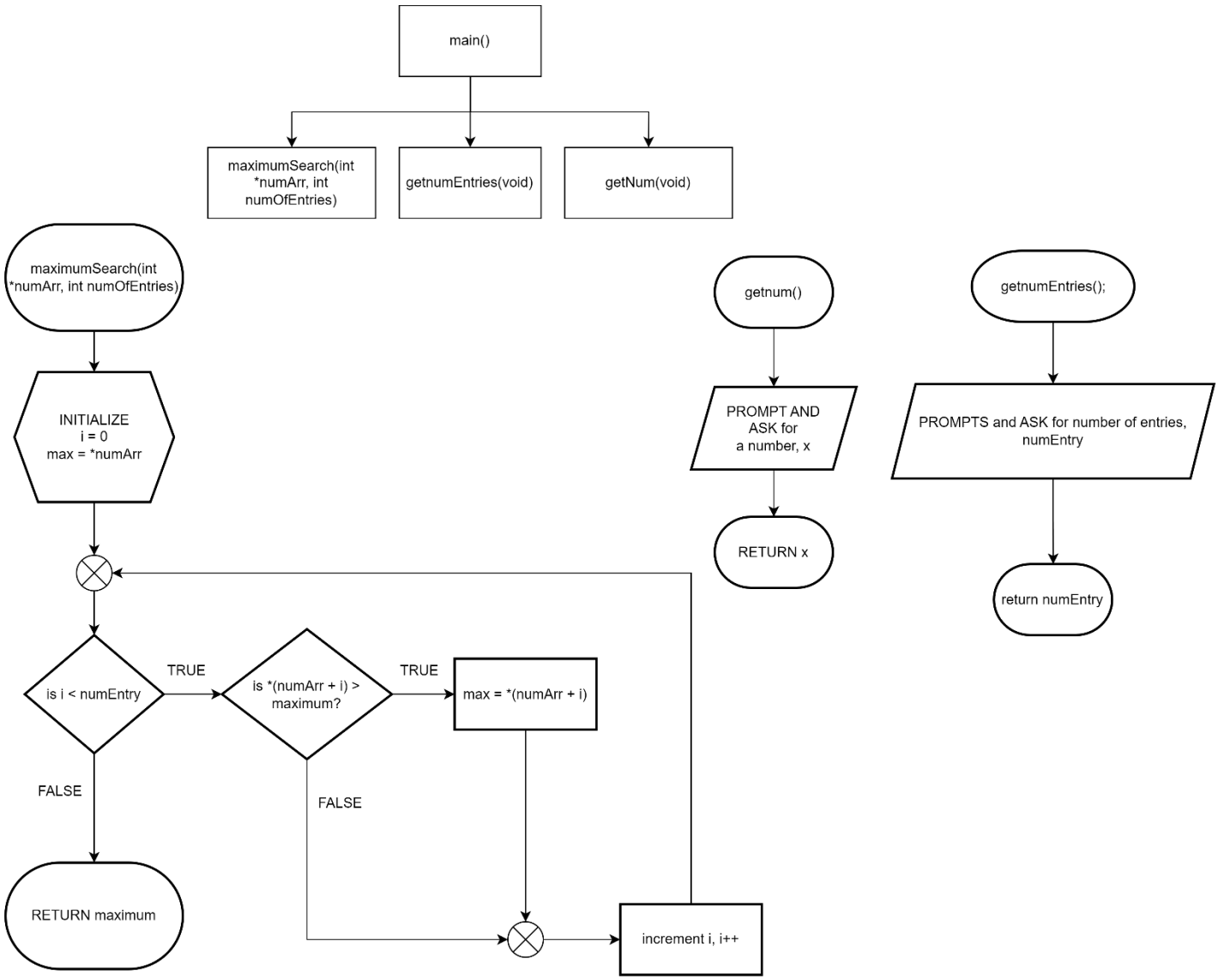

main()
getSearched(void);
indexSearch(int *numArr, int numEntry, int numToSearch)
getnumEntries(void);
getNum(void);

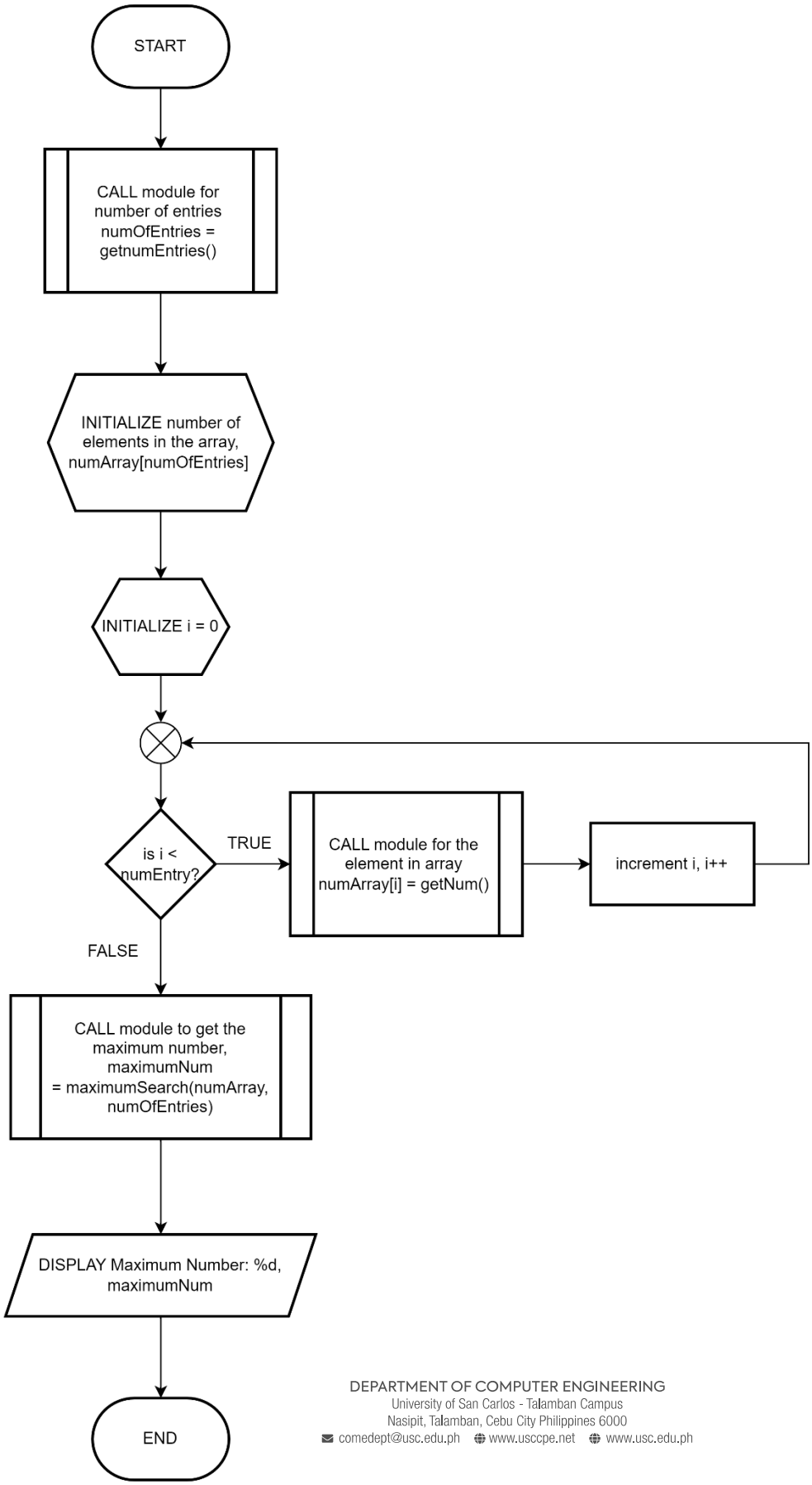

indexSearch(int *numArr, int numEntry, int numToSearch)
INITIALIZE i = 0, index = -1
increment i, i++
is i < numEntry AND index < 0?
True
IS *(numArr + i) == numToSearch?
True
index = i
False
False
RETURN index


getnumEntries();
PROMPTS and ASK for number of entries, numEntry
return numEntry
getnum()
PROMPT AND ASK for a number, x
RETURN x
getSearched()
PROMPT AND ASK for number to be searched, numSearched
RETURN numSearched


START
CALL module for number of entries numEntry = getnumEntries()
INITIALIZE number of elements in the array, numArray[numEntry]
INITIALIZE i = 0
is i < numEntry?
TRUE
CALL module for the element in array numArray[i] = getNum()
increment i, i++
FALSE
CALL module to get the number to be searched, numSearched = getSearched()
CALL module to find index where number is located, indexLoc = indexSearch(numArray, numEntry, numSearched)
if indexLoc >= 0
TRUE
DISPLAY "found in index %d", indexLoc
FALSE
DISPLAY "Not found"
END

3. Maximum Number.
   A function that determines and returns the maximum number in an array of integers using pointers. The function accepts the starting address of the array and number of entries.




START
CALL module for number of entries numOfEntries = getnumEntries()
INITIALIZE number of elements in the array, numArray[numOfEntries]
INITIALIZE i = 0
is i < numEntry?
TRUE
CALL module for the element in array numArray[i] = getNum()
increment i, i++
FALSE
CALL module to get the maximum number, maximumNum = maximumSearch(numArray, numOfEntries)
DISPLAY Maximum Number: %d, maximumNum
END