Birzeit University - Faculty of Engineering and Technology
Electrical & Computer Engineering Department - ENCS4330
Real-Time Applications & Embedded Systems - 2^{nd} semester - 2023/2024

Project #2
Interprocess communication techniques under Linux
Due: April 30, 2024

Instructor: Dr. Hanna Bullata

Distributing wheat flour aid bags to our people in northern Gaza strip simulation

We would like to create a multi-processing application that simulates the behavior of a full wheat flour distribution operation to our people in northern Gaza who are subject to starvation by the zionist occupation entity. The objective of the simulation is to check on the necessary conditions to overcome starvation within the scope described below.

Relief committee workers help in collecting, splitting and distributing the wheat flour containers that are parachuted from cargo planes to families in need. As such, assume relief committee workers are composed of 3 categories:

- Collecting wheat flour relief workers.
- Splitting wheat flour relief workers.
- Distributing wheat flour relief workers.

The application can be explained as follows:

- A user-defined number of cargo planes will parachute wheat flour containers over northern Gaza. Each cargo plane has a random number of wheat flour containers that is bound to a user-defined range and and will drop each flour container every random amount of time that is bound to a user-defined period. Once cargo planes drop all their containers, they leave northern Gaza airspace and go for a re-fill. The re-fill period must belong to a user-defined range.

  Of course, while cargo planes fly over northern Gaza, they must avoid collision with each other.

- Occupation forces will do their best to fire ground-air missiles on these containers and make them explode while dropping to the ground. If wheat flour containers explode too high in the air, the content is totally lost. The nearer to the ground these containers are hit, the better the chance to collect part of the content. If lucky enough, some containers might land safely on the ground. Add user-defined arguments as needed to describe that fact.

- A user-defined number of collecting relief committees on the ground are responsible to collect the wheat flour containers and take them to a safe storage area. Assume each collecting relief committee is composed of a user-defined number of relief workers. A single relief committee is needed to handle one wheat flour container at a time.

- Once a container is taken to the safe storage area, a collecting relief committee can go collect another container. Collecting relief committee workers get tired with time while more containers get collected. Sadly enough, tired collecting relief committee workers can be martyred by occupation forces snipers and the chances are higher the more they get tired. Whenever a collecting relief worker is killed, a splitting relief worker will join the collecting group as a replacement to the deceased worker.

- In safe storage areas, splitting relief committee workers split wheat flour containers into 1-Kg bags to be distributed for families in need.
- Distribution relief workers are responsible to distribute the wheat flour bags. Each distribution relief worker can handle a user-defined number of bags per distribution trip. Priority is always given to the families whose starvation rate is the highest. When the wheat flour bags have been distributed by a distribution relief worker, he can go back to the storage area to get more bags. Sadly enough, tired distribution relief workers can be martyred by occupation forces snipers during distribution trips and the chances are higher the more they get tired. If the number of distributing relief worker drops below a user-defined number, some splitting relief workers might switch to become distributing relief workers as a replacement.
- Starvation level for the families increase with time while waiting for wheat bags and drops only when a wheat flour bag is received. Families suffering from starvation are subject to passing away if wheat flour bags are not received on time. This is why priority must be given to families who are in deep need.

  Assume families locations are fixed. As such, families do not move from location to location during the simulation.
- The simulation ends if any of the following is true:
  - The application has been running for more than a user-defined threshold time and the family death rate is below a user-defined threshold.
  - The number of cargo planes that crashed by collision in northern Gaza airspace exceeds a user-defined threshold.
  - The number of wheat flour containers who have been shot down exceeds a user-defined threshold.
  - The number of collecting relief committee workers who have been martyred is above a user-defined threshold.
  - The number of distributing relief committee workers who have been martyred is above a user-defined threshold.
  - The number of families who passed away due to starvation is above a user-defined threshold.

What you should do

- In order to implement the above-described application, you can choose any combination of IPC techniques (message queues, semaphores, shared memory) in addition to pipes, fifos & signals. Of course you don't need to use all of them. Just pick the one(s) that serve the purpose of the application.
- Feel free to add more details to the above description to make it as realistic as possible.
- Write the code for the above-described application using a multi-processing approach. Create a makefile that will manage the application build process.
- In order to avoid hard-coding values in your programs, think of creating a text file that contains all the values and ranges that should be user-defined and give the file name as an argument to the main program. That will spare you from having to change your code permanently and re-compile.
- Use graphics elements from opengl library in order to best illustrate the application. Nothing fancy, just simple and elegant elements are enough.
- Test your program.

- Check that your program is bug-free. Use the `gdb` debugger in case you are having problems during writing the code (and most probably you will :-). In such a case, compile your code using the `-g` option of the `gcc`.
- Send the zipped folder that contains your source code and your executable before the deadline. If the deadline is reached and you are still having problems with your code, just send it as is!