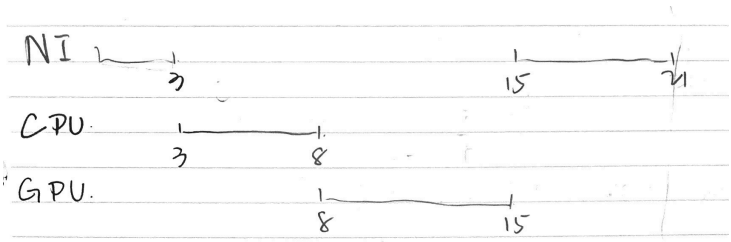
Problem 1



a).

$U_{NI} = \frac{3+6}{21} = \frac{9}{21} = \frac{3}{7}$

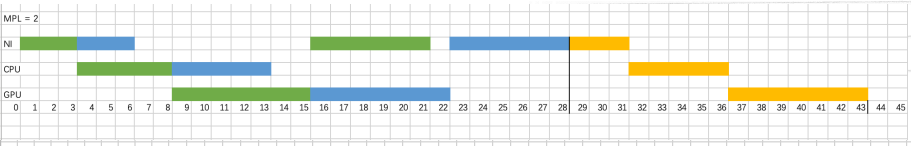
c). bottleneck system is NI.

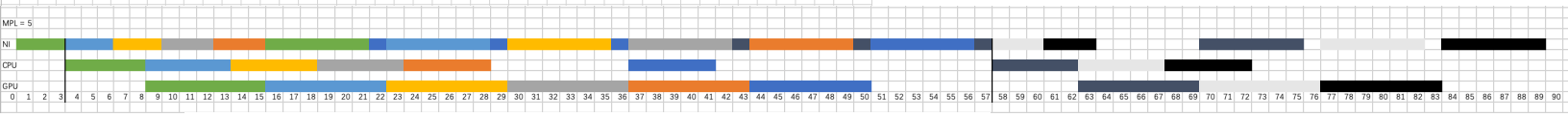
$U_{CPU} = \frac{5}{21}$

$U_{GPU} = \frac{7}{21} = \frac{1}{3}$

b). pattern length = 21

Throughput = $\frac{1}{21}$





d).

$U_{NI} = \frac{18}{27}$ $U_{CPU} = \frac{10}{27}$ $U_{GPU} = \frac{14}{27}$

e).

throughput = $\frac{2}{27}$

f).

MLP = 5

$\frac{54}{54}$ $\frac{30}{54} = \frac{5}{9}$ $\frac{42}{54} = \frac{7}{9}$

g) capacity = max throughput = $\frac{6}{54} = \frac{1}{9}$

Problem 2

The Gidiyor-Gitti company provides zero-contact through-the-window catapult-aided food delivery. Each time a new delivery order is placed, the system needs to compute (1) wind speed at the destination address for catapult calibration and (2) traffic conditions for the driver to pick up the food to be catapulted. These pieces of information are processed in a computing cluster. On average, each order triggers computation that is completed in 5.5 minutes. 40% of that time is spent on pure wind estimation at the CPU, while 25% is spent on I/O. As you are trying to shave some time off the current order-to-catapult time, you narrow down two possible optimizations. A first option (A) consists in upgrading the CPUs. The upgrade will cost $10,000. The new processors will take 20% less time to process each instruction (e.g., instruction cycle goes down from 10 nanoseconds to 8 nanoseconds). The second option (B) is to use a new array of high-performance I/O devices that brings down the latency of each I/O operation from 12 microseconds to 10 microseconds (which costs $35,000). Answer the following questions:

a) What speedup do you expect if you pick option A?

b) What speedup do you expect if you pick option B?

c) On a per-dollar basis, which of the two improvements is better?

d) What is the speedup that you expect if you pick both improvements?

e) What is the theoretical limit for the speedup that can be achieved by making the CPUs infinitely fast?

f) What is the theoretical limit for the speedup that can be achieved by making the I/O devices infinitely fast?

5.5 → 40% wind estimation (CPU). 2.2 min. A: $2.2(0.2) = 0.44$ min.

→ 25% I/O. 1.375 min. B: $1.375\left(\frac{12-10}{12}\right) = 0.229$ min.

a). $\text{Speedup}_A = \frac{5.5}{5.5-0.44} = 1.0870$

b). $\text{Speedup}_B = \frac{5.5}{5.5-0.229} = 1.0434$

c). $\frac{1.0870}{10000} > \frac{1.0434}{35000}$, so A.

A B

d). $\frac{5.5}{5.5-0.44-0.229} = 1.1385$

e). $\frac{1}{1-0.4} = 1.6667$

f). $\frac{1}{1-0.25} = 1.3333$

Problem 3

A scientific large-scale application that performs weather pattern prediction is deployed on a 100-core machine—i.e., a machine with 100 CPUs. To save power, some CPUs can be turned off by the system. In which case, the application is restricted to run only on the CPUs that are powered on (online). In a single run, the application performs the following sequence of operations. (1) it initializes its state which takes 5 seconds and cannot be parallelized; (2) it launches one prediction heuristic per each square-foot of covered area. These are kept independent from each other and thus can be executed in parallel, with each taking 1 second to complete. The total area covered by the weather prediction is 100 square feet in size. (3) It serializes the result obtained from each local prediction into a global prediction. This step takes 15 seconds and cannot be parallelized. Answer the following and motivate your answers. $\frac{120}{120} = 1$

a) What is the speed-up of the entire application when it operates on a single CPU, compared to the case where it operates on all the available CPUs? $5+1+15=21$ s → all 100 CPUs. $\frac{120}{21} = 5.714$

b) How many CPUs we would need to keep online to ensure that we are able to achieve a weather prediction throughput of 2 predictions per minute? speedup $= \frac{2(120)}{60} = 4$ ✓

$2(5+\frac{100}{N}+15)=60$.

$5+\frac{100}{N}+15=30$.

$\frac{100}{N}=20$.

$N=10$.

c). $5+1+15=21s=\frac{21}{60}$ min.

$\rightarrow \frac{1}{\frac{21}{60}} = 2.857$ predictions/min.

d) No because Amdahl is a very ideal case, it assumes infinite CPUs, but if we have only 30, the split of the work is hard to achieve equally nicely, for example, if two block need to be together, this might be a problem when we assign the work to CPUs. Differently, if only one CPU is utilized, we won't have this problem.

e) It is beneficial, because we can use the turbo–boosting CPU to run the part that can not be parallelized (1) & (3), and continued to utilized it for the part that can be parallelized (2). Therefore, the runtime would be (5+15)/2 + 100/51 = 12s which is faster than 21s. Therefore, the speedup would increase, as well as the throughput.