1. There are $m = 6n\ln(n)$ jobs assigned at random to n processors (note: remember $\ln(x) = \log_e(x)$). Consider a processor p and show, with high probability in parameter n, processor p does not receive more than $12\ln(n)$ jobs. (Hint: define an appropriate indicator variable for each job and apply the following Chernoff bound).

Theorem 1. *Siano $X_1, X_2, \ldots, X_n$ variabili indicatore indipendenti con $E[X_i] = p_i, 0 < p_i < 1$. Sia $X = \sum_{i=1}^{n} X_i$ e $\mu = E[X]$. Allora, per ogni $0 < \delta \leq 1$,*

$$Pr(X > (1+\delta)\mu) \leq e^{-\frac{\delta^2\mu}{3}}.$$

Here, we have X_i with $i = 1, 2 \ldots 6n\ln(n)$, an indicator variable with value 1 if the i^{th} job gets assigned to processor p. Considering all events X_i are independent, so we get $\Pr(X_i = 1) = \frac{1}{n}$.

Specifically, $X = \sum_{i=1}^{6n\ln(n)} X_i$. Given the processor has to receive no more than $12\ln(n)$ jobs, this is our bound and in specific we have $12\ln(n) = (1+\delta)\mu$:

$$\Pr(X > 12\ln(n))$$

Now, we want to solve for δ:

$$12\ln n = 6\ln n(1+\delta)$$

$$\frac{12\ln n}{6\ln n} = 1+\delta$$

$$2 = 1+\delta$$

$$\delta = 1$$

Then, we substitute everything:

$$\Pr(X > (1+1)\mu) = \Pr(X > 12\ln n) \leq e^{-\frac{1^2\cdot\mu}{3}}$$

$$\Pr(X > 12\ln n) \leq e^{-\frac{6\ln n}{3}} = e^{-2\ln n} = \left(e^{\ln n}\right)^{-2} = (n)^{-2} = \frac{1}{n^2}$$

Thus we have:

$$\Pr(X > 12\ln n) \leq \frac{1}{n^2}$$

which means:

$$\Pr(X \leq 12\ln n) \geq 1 - \frac{1}{n^2}$$

This shows that with high probability, the processor p does not receive more than $12\ln(n)$ jobs

2.

Exercise 2 (9 points) For $n \gg 1$, let $X_1, X_2, \ldots, X_n$ be independent indicator random variables such that $Pr(X_i = 1) = (6\ln n)/n$ (recall that $\ln n = \log_e n$). Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. By applying the following Chernoff bound

$$Pr(X > (1+\delta)\mu) < e^{-\mu\delta^2/2} \quad \text{for } 0 < \delta \leq 2e-1$$

prove that

$$Pr(X > 10\ln n) < \frac{1}{n^c}$$

for some positive constant c to be determined.

Here, given $\Pr(X_i = 1) = \frac{6\ln(n)}{n}$, we have:

$$\mu = E[X] = \sum_{i=1}^{n} E[X_i] = n \cdot \frac{6\ln n}{n} = 6\ln n$$

We then set the Chernoff bounds to be:

$$10\ln(n) = (1+\delta)6\ln(n)$$

Solve for δ:

$$\frac{10\ln n}{6\ln n} = 1+\delta$$

$$\frac{10}{6} = 1+\delta$$

$$\frac{5}{3} = 1+\delta$$

$$\delta = \frac{5}{3} - 1 = \frac{2}{3}$$

Apply the Chernoff bound:

$$\Pr(X > 10\ln n) = \Pr(X > (1+\frac{2}{3})\mu)$$

Using the Chernoff bound formula:

$$\Pr(X > (1+\delta)\mu) < e^{-\frac{\mu\delta^2}{2}}$$

Substituting $\delta = \frac{2}{3}$ and $\mu = 6\ln n$:

$$\Pr(X > (1+\frac{2}{3})6\ln n) < e^{-\frac{6\ln n\cdot(\frac{2}{3})^2}{2}}$$

$$= e^{-\frac{6\ln n\cdot\frac{4}{9}}{2}}$$

$$= e^{-\frac{6\ln n\cdot\frac{4}{9}}{2}}$$

$$= e^{-\frac{24\ln n}{18}}$$

$$= e^{-\frac{4\ln n}{3}}$$

$$e^{-\frac{4\ln n}{3}} = (e^{\ln n})^{-\frac{4}{3}} = n^{-\frac{4}{3}}$$

Thus, we have:

$$\Pr(X > 10\ln n) < \frac{1}{n^{4/3}}$$

Therefore, we have correctly proved that:

$$\Pr(X > 10\ln(n)) < \frac{1}{n^{\frac{4}{3}}}$$

for $c = \frac{4}{3}$, bounding correctly for a positive constant c.

3.

Exercise 2 (11 points) Let S be a set of n distinct positive integers, and let WORK(S) be a procedure which, given input S, returns an integer by performing n^2 operations. Now consider the following randomized algorithm:

```
RAND_REC(S)
   if |S| <= 1 then return 1
   x = WORK(S)
   p = RANDOM(S)
   S1 = {s in S such that s < p}
   S2 = {s in S such that s > p}
   if (|S1| >= |S2|) then
      y = RAND_REC(S1)
   else
      y = RAND_REC(S2)
   return x + y
```

Applying the following Chernoff bound show that the complexity of RAND_REC(S) is $O(n^2 \log n)$ with high probability. (Hint: recall the analysis of randomized QuickSort.)

Theorem 1. *Let $X_1, X_2, \ldots, X_n$ be independent indicator random variables such that $E[X_i] = p_i, 0 < p_i < 1$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then, for $0 < \delta \leq 1$,*

$$\Pr(X < (1-\delta)\mu) < e^{-\mu\delta^2/2}.$$

The algorithm basically partitions the original sets into two subsets S_1 and S_2 containing elements from both sets, then calling the procedure on the larger set and returning the sum. We observe the following:

- if the sets are balanced S_1 and S_2, then $O(\log(n))$
- we want to ensure that each recursive call reduces the size of the set by a significant fraction
    - then consider a good pivot ensuring a probability of $\max(|S_1|, |S_2|) \leq \frac{3n}{4}$

Using Chernoff bounds, we can show that the probability of having a "bad" pivot that doesn't split the set significantly decreases exponentially. For a random pivot p, the probability that the size of the larger partition the size of the larger partition is at most $\frac{3n}{4}$ can be derived as follows:

- Let X be the size of the larger partition.
- For $0 < \delta \leq 1$, we have:

$$\Pr\left(X > (1+\delta)\frac{n}{2}\right) \leq e^{-\frac{\frac{n}{2}\delta^2}{3}}$$

We want to ensure that the size of the larger partition is less than $\frac{3n}{4}$:

- Set $(1+\delta)\frac{n}{2} = \frac{3n}{4}$

$$1+\delta = \frac{3}{2} \implies \delta = \frac{1}{2}$$

Using this value in the Chernoff bound:

- $\mu = \frac{n}{2}$

$$\Pr\left(X > \frac{3n}{4}\right) \leq e^{-\frac{\frac{n}{2}\left(\frac{1}{2}\right)^2}{3}} = e^{-\frac{n}{24}}$$

This shows that the probability of a "bad" split is exponentially small.

The total work done at each level of recursion, $WORK(S)$ performs n^2 operations.

Total work done is:

$$\sum_{i=0}^{O(\log(n))} n^2 = O(n^2 \log(n))$$

4.

Exercise 2 (9 points) Suppose you toss $n \gg 1$ times a coin: applying the following Chernoff bound show that the probability that you obtain more than $n/2 + \sqrt{6n \ln n}/2$ heads is at most $1/n$.[1]

Theorem 1. *Let $X_1, X_2, \ldots, X_n$ be independent indicator random variables such that $E[X_i] = p_i, 0 < p_i < 1$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then, for $0 < \delta \leq 1$,*

$$\Pr(X > (1+\delta)\mu) \leq e^{-\mu\delta^2/3}.$$

[1] Recall that $\ln n = \log_e n$.

Let X_i be an indicator random variable for the i-th coin toss, where $X_i = 1$ if the outcome is heads and $X_i = 0$ if the outcome is tails.

Since the coin is fair, $\Pr(X_i = 1) = \frac{1}{2}$. Therefore,

$$\mu = \mathbb{E}[X] = \sum_{i=1}^{n} \mathbb{E}[X_i] = n \cdot \frac{1}{2} = \frac{n}{2}$$

We need to find δ such that $\frac{n}{2}+\sqrt{6n\ln\left(\frac{n}{2}\right)}=(1+\delta)\frac{n}{2}$.

1. **Solving for δ:**

$$\frac{n}{2}+\sqrt{6n\ln\left(\frac{n}{2}\right)}=(1+\delta)\frac{n}{2}$$

$$1+\frac{2\sqrt{6n\ln\left(\frac{n}{2}\right)}}{n}=1+\delta$$

$$\delta=\frac{2\sqrt{6n\ln\left(\frac{n}{2}\right)}}{n}$$

Simplifying δ:

$$\delta=2\sqrt{\frac{6\ln\left(\frac{n}{2}\right)}{n}}$$

We substitute $\delta=2\sqrt{\frac{6\ln\left(\frac{n}{2}\right)}{n}}$ and $\mu=\frac{n}{2}$ into the Chernoff bound:

$$\Pr\left(X>\frac{n}{2}+\sqrt{6n\ln\left(\frac{n}{2}\right)}\right)=\Pr\left(X>(1+\delta)\mu\right)\leq e^{-\frac{\mu\delta^2}{3}}$$

Substituting μ and δ:

$$e^{-\frac{\frac{n}{2}\left(2\sqrt{\frac{6\ln\left(\frac{n}{2}\right)}{n}}\right)^2}{3}}$$

$$=e^{-\frac{\frac{n}{2}\cdot\frac{24\ln\left(\frac{n}{2}\right)}{n}}{3}}$$

$$=e^{-\frac{12\ln\left(\frac{n}{2}\right)}{3}}$$

$$=e^{-4\ln\left(\frac{n}{2}\right)}$$

Since $\ln\left(\frac{n}{2}\right)=\ln n-\ln 2$:

$$e^{-4(\ln n-\ln 2)}=e^{-4\ln n+4\ln 2}$$

Simplifying further:

$$=e^{-4\ln n}\cdot e^{4\ln 2}$$

$$=(e^{\ln n})^{-4}\cdot 2^4$$

$$=n^{-4}\cdot 16$$

Finally, we get:

$$\Pr\left(X>\frac{n}{2}+\sqrt{6n\ln\left(\frac{n}{2}\right)}\right)\leq\frac{16}{n^4}$$

For $n\gg 1$, $\frac{16}{n^4}<\frac{1}{n}$, so we have shown that:

$$\Pr\left(X>\frac{n}{2}+\sqrt{6n\ln\left(\frac{n}{2}\right)}\right)\leq\frac{1}{n}$$

5.

Exercise 2 (10 points) Suppose you have a randomized algorithm for a minimization problem A that returns the correct output with probability at least $1/n$, where n is the input size. Show how to obtain an algorithm for A that returns the correct output with high probability. (Hint: for the analysis use this inequality: $(1+x/y)^y \leq e^x$ for $y \geq 1$, $y \geq x$.)

Characterize the event:

$$\Pr\left(X > \frac{1}{n}\right) > \frac{1}{n^d}$$

We want to find a value for k such that $Pr\left(1-\frac{1}{n}\right)^k \leq \frac{1}{n^d}$. In this case, it's standard the use of this inequality:

$$\left(1+\frac{x}{y}\right)^y \leq e^x, y \geq 1, y \geq x$$

Recall the following:

$$e^{-\ln(n^d)} = \frac{1}{n^d}$$

This inequality is derived from the exponential function and the binomial expansion. It represents an upper bound on the expression $\left(1+\frac{x}{y}\right)^y$, showing that it grows slower than e^x. The probability of not contracting the minimum cut in each iteration needs to be bounded and manipulated to ensure the overall algorithm's success probability is high.

Recall the following:

$$e^{-\ln(n^d)} = \frac{1}{n^d}$$

By choosing $k = \frac{dn}{2}\ln(n)$ it follows that:

$$\left(1-\frac{1}{n^2}\right)^{k=n} \leq e^{-1} = \frac{1}{e} \rightarrow \text{is } \underline{\text{not}} \text{ in the form } \frac{1}{n^d}$$

$$\left(\left(1-\frac{1}{n}\right)^n\right)^{\ln(n^d)} = \left(1-\frac{1}{m}\right)^{n\ln(n^d)}$$

Let's wrap up:

$$\left(1-\frac{1}{n}\right)^{k=dn\ln(n^d)} = \left(\left(1-\frac{1}{m}\right)^n\right)^{\ln(n^d)}$$

$$\leq (e^{-1})^{\ln(n^d)} = e^{-\ln(n^d)} = \frac{1}{n^d}$$