Problem type 1:

Provide the regular expression for the following language:

(See variants below)

Assume $\Sigma = \{0, 1\}$

a. **BYH**

All strings containing the *substring* **000**

Solution: $(0+1)^* 000 (0+1)^*$ ■

b. **BYA**

All strings containing the *subsequence* **000**

Solution: $(0+1)^* 0 (0+1)^* 0 (0+1)^* 0 (0+1)^*$ ■

c. **BYC**

All strings that do not contain the *substring* **00**.

Solution: Basically we simply can't have two zeros appear consecutively. Every zero must have a **1** after it. So we can construct a regular expression that does this: $(\varepsilon + 0)(1(\varepsilon + 0))^*$. See Lab1P2 for a more detailed explanation. ■

d. **BYG**

All strings that do not contain the *subsequence* **00**

Solution: This is just another way of saying that the strings must have at most one **0**. $1^*(\varepsilon + 0)1^*$ ■

e. **BYF**

All strings where every run of **0**'s is a multiple of 2.

Solution: $\left(1 + (00)^*\right)^*$ ■

f. **BYE**

All strings were every run of **0**'s is *not* a multiple of two.

Solution: Another way to say this is that every run of is odd. So let's start off with something like $\left(1^+ 0 (00)^*\right)^*$ (we need the "+"). But then you have to account for starting on zero and starting/ending on a run of ones: $1^* 0 (00)^* \left(1^+ 0 (00)^*\right)^* 1^*$.

As long as you got something along the lines of $\left(1^+ 0 (00)^*\right)^*$, we'll give you full credit. ■

g. **BYB**

All string containing at **least** three $\mathtt{0}$'s

Solution: $(\mathtt{0}+\mathtt{1})^*\mathtt{0}(\mathtt{0}+\mathtt{1})^*\mathtt{0}(\mathtt{0}+\mathtt{1})^*\mathtt{0}(\mathtt{0}+\mathtt{1})^*$ ■

h. **BYD**

All string containing at **most** three $\mathtt{0}$'s

Solution: $\mathtt{1}^*(\varepsilon+\mathtt{0})\mathtt{1}^*(\varepsilon+\mathtt{0})\mathtt{1}^*(\varepsilon+\mathtt{0})\mathtt{1}^*$ ■