Randomized algorithms

Exercise 1  Schwartz Zippel lemma, ∼1980 (and DeMillo Lipton, ∼1978?)

Let $F$ be a field.

1. Prove the so-called Schwartz-Zippel lemma:
   
   If $p \in F[x_1, \ldots, x_n]$ is non zero, and has total degree $d$ (maximum degree $\nu_1 + \ldots + \nu_n$ of a monomial $x_1^{\nu_1} \cdots x_n^{\nu_n}$), if $S \subset F$, and $r_1, \ldots, r_n$ are uniformly and independently picked at random in $S$, then $\Pr[p(r_1, \ldots, r_n) = 0] \leq d |S|^{-1}$. 

2. Deduce a randomized algorithm from last question, to test if a polynomial is zero, for an infinite field. When does it make mistakes?

3. In the case where the polynomial may be sparse ("creux" in French), it is represented as an algebraic circuit — with variables in the leaves, and operations $+, -, \times$. Supposing the circuit has size $m$, bound its degree, and give an estimate of the expected arithmetic cost (in terms of operations in $F$) of deciding if it is zero.

4. Estimate the worst bit size of an intermediate value in your algorithm in the case where $p \in \mathbb{Z}[x_1, \ldots, x_n] \subseteq \mathbb{Q}[x_1, \ldots, x_n]$. 
   (this intermediate expressions' "swell" is typical in formal calculus)

5. Any idea to tackle this?

Exercise 2  One-sided error

Recall that $\text{RTIME}(T(n))$ is the set of languages $L$ for which there is a probabilistic TM running in time $T(n)$ w.r.t. $x \in L \implies \Pr[M(x) = 1] \geq 2/3$ and $x \notin L \implies \Pr[M(x) = 1] = 0$. Recall also $\text{RP} := \cup_{c>0} \text{RTIME}(n^c)$.

Let also $\text{ZTIME}(T(n))$ be the set of languages $L$ for which there is a TM $M$ running in expected time $O(T(n))$ such that for any $x$, it decides exactly $x \in L$ when it returns. Define now $\text{ZPP} := \cup_{c>0} \text{ZTIME}(n^c)$. Prove that $\text{ZPP} = \text{RP} \cap \text{coRP}$.

Exercise 3  Primality testing

The aim of this exercise is to give an overview of an efficient algorithm for primality testing — answering the question whether $N$ is prime, in polynomial time in $\log(N)$. It dates back to the 70s.

For $1 \leq A \leq N$, let $QR_N(A)$ be defined as:

- $0$ if $\gcd(A, N) \neq 1$,
- $1$ otherwise, when there exists a $B$ co-prime with $N$, such that $A = B^2 \mod (N)$,
- $-1$ otherwise.

We will rely on the following facts (almost all admitted):

- for $N$ an odd prime, and $1 \leq A < N$, $QR_N(A) = A^{(N-1)/2} \mod (N)$,
- the Jacobi symbol $\left( \frac{N}{A} \right) := \prod_{p \text{ prime dividing } N} QR_p(A)$ can be computed in time $O(\log(N) \log(A))$,
- for $N$ composite, at most half of the $A$s such that $\gcd(A, N) = 1$ satisfy $\left( \frac{N}{A} \right) = A^{(N-1)/2} \mod (N)$.

1. Prove the first admitted item.
2. Use these facts to give an efficient randomized algorithm for testing primality. In which class is it?