2 Foundations of Computer Science (avsm2)

(a) A prime number is a natural number greater than 1 that has no positive divisors other than 1 and itself. We wish to implement a primality test in OCaml that checks if a positive input integer is prime. A simple primality test is via trial division: given an input positive integer \( n \), check if it is divisible by any prime number between 2 and \( \sqrt{n} \). For any divisor \( p \geq \sqrt{n} \), there must be another divisor \( n/p \leq \sqrt{n} \), and a prime divisor \( q \) of \( n/p \), and therefore looking for prime divisors where \( p \leq \sqrt{n} \) is sufficient.

Define a function `is_prime` which accepts a positive input integer and returns a boolean to indicate if it is prime or not. To simplify your code, you can avoid calculating square roots by checking for prime divisors where \( p^2 \leq n \). You can assume the existence of a `(mod)` operator which returns the integer remainder of two integers. For example, \( 3 \mod 2 \) will return 1. The type definitions are:

```ocaml
val (mod): int -> int -> int
val is_prime : int -> bool
```

(b) In functional programming, `fold` functions are higher order functions that process data structure elements in order and build a return value.

(i) `fold_range a b f acc` is a specialised integer fold that applies \( f(n) \) where \( b \leq n \leq a \), with initial value \( acc \). For example, `fold_range 1 3 (+) 10` would return 16 (from \( 10+1+2+3 \)). Define this function with the type:

```ocaml
val fold_range: int -> int -> (int -> 'a -> 'a) -> 'a -> 'a
```

(ii) `fold f acc l` is a more generic function that applies \( f \) over a list \( l \) with initial value \( acc \). Define this function with the type:

```ocaml
val fold: ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a
```

(iii) Explain the time and space complexity of your implementation of `fold` and briefly discuss whether it is tail recursive or not.

(c) `all_primes a b` is a function that returns all of the valid prime numbers between \( a \) and \( b \) inclusively. Define this function and its associated type, using the previous definitions of `fold_range` and `is_prime`.

[8 marks]