*School of Software Engineering*

UNIVERSITYOF SCIENCE AND TECHNOLOGY OF CHINA

**Compilers, Spring 2024**

**Quiz 4: Type Checking**

Name: Id:

1. (10 points) Given the following grammar for a small subset of the C programming language:

***prog* -> *decs stm***

***decs* -> *type* ID; *decs***

**| .**

***type* -> int | bool**

***stm* -> if (*exp*) *stm* else *stm***

**| while (*exp*) *stm***

***| ID = exp***

***exp* -> (*exp*) | *exp*+*exp* | exp<exp | *exp*&&*exp***

**| ID | NUM | TRUE | FALSE**

The followings are the type checking rules for this language. Write down some code to finish the type checking function check\_exp(). (You might use pseudo-code.)

In the next, the context \Gamma ::= t1 x1, ..., tn xn, where ti are types and xi are variables, for 1<=i<=n.

 

 

 

 

**type check\_exp (Gamma g, exp e) =**

**// Your code here:**