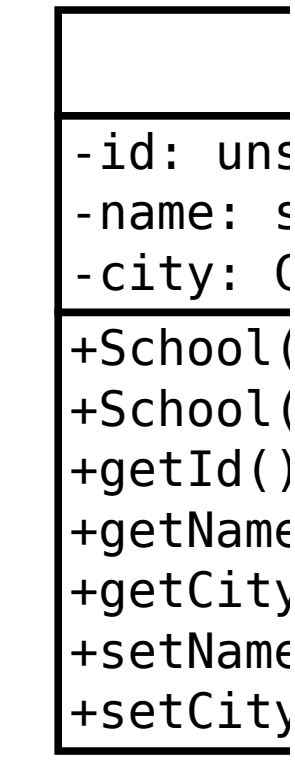
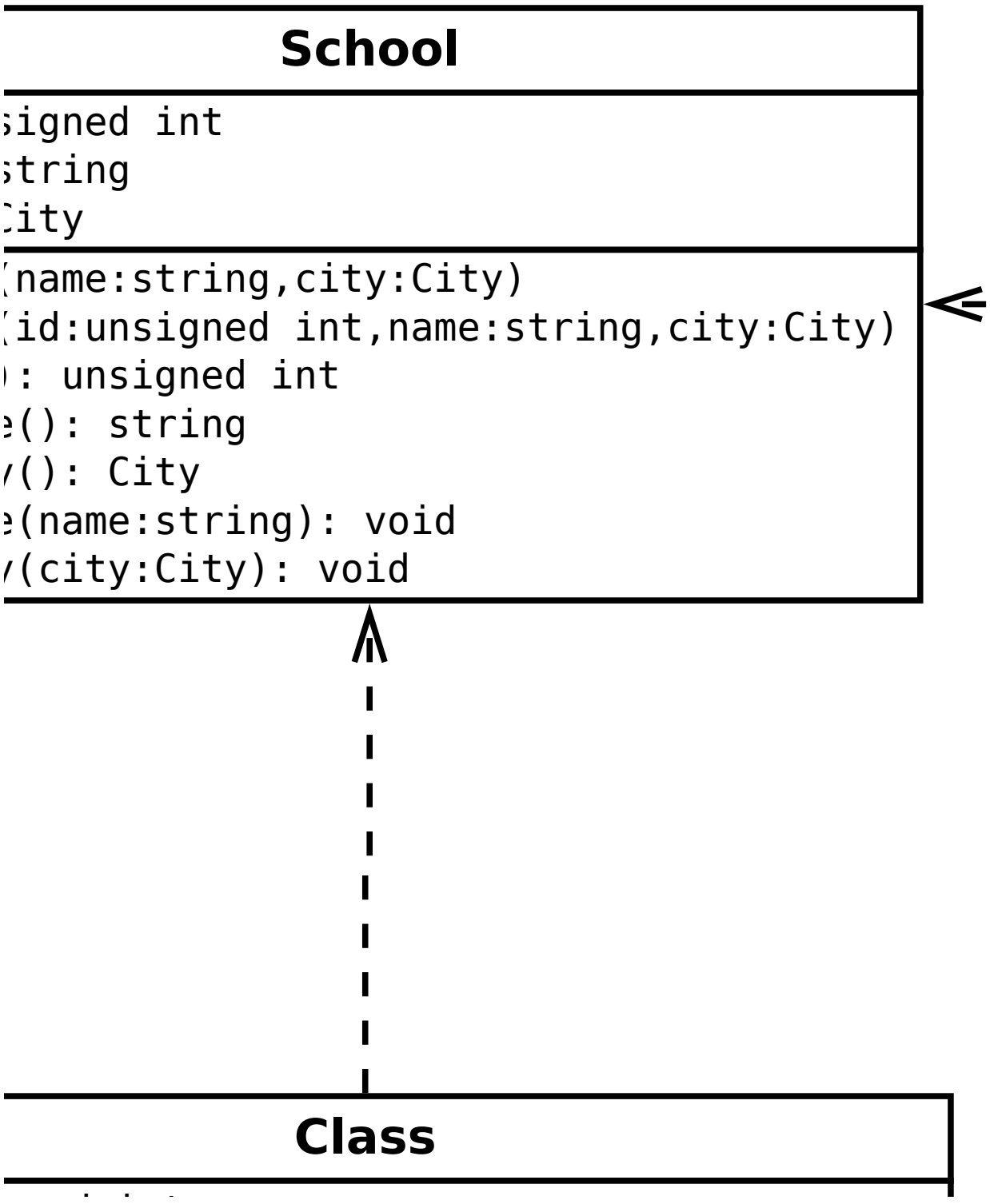
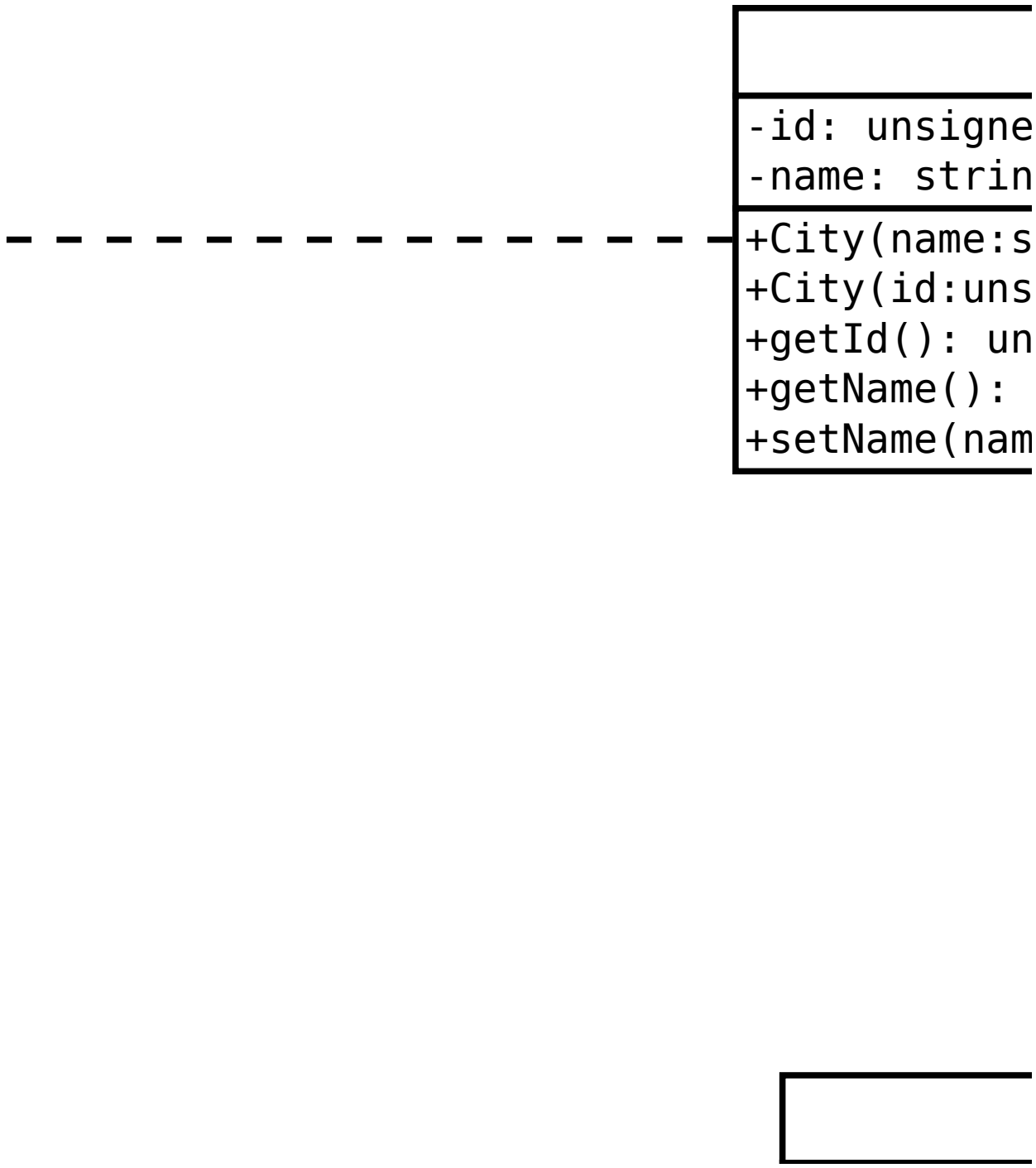
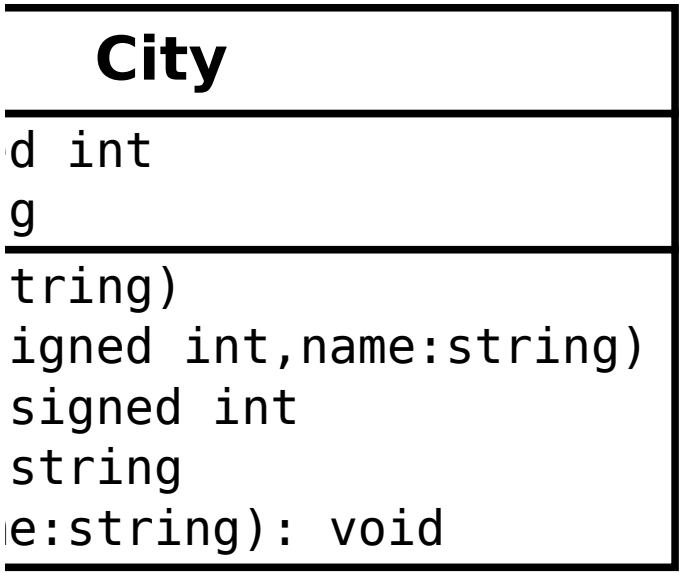
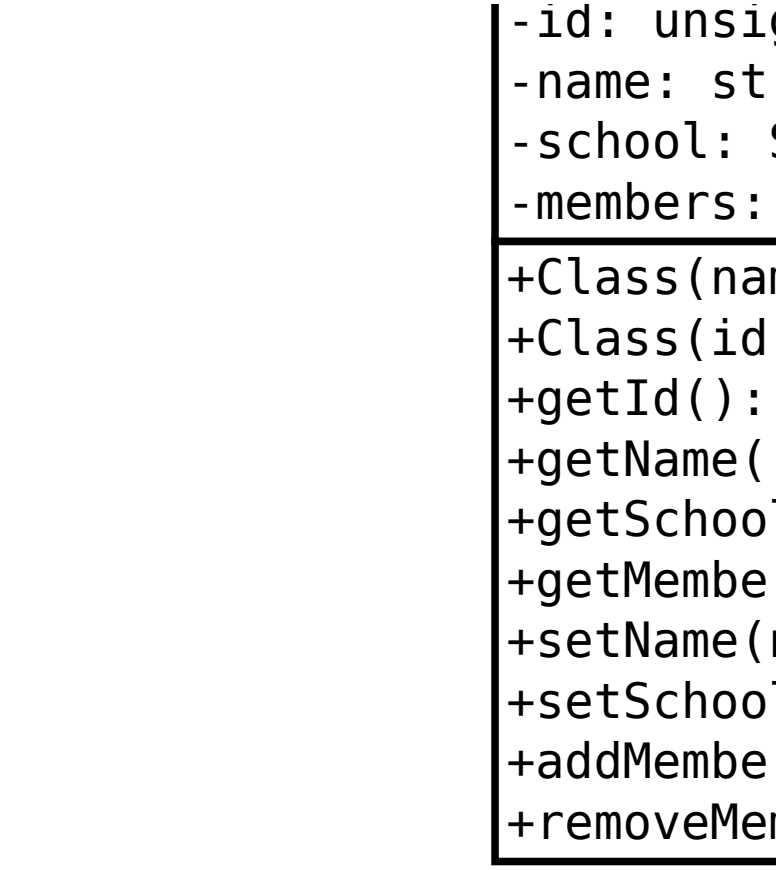

-id: uns
-name: s
-city: C
+School(
+School(
+getId()
+getName
+getCity
+setName
+setCity

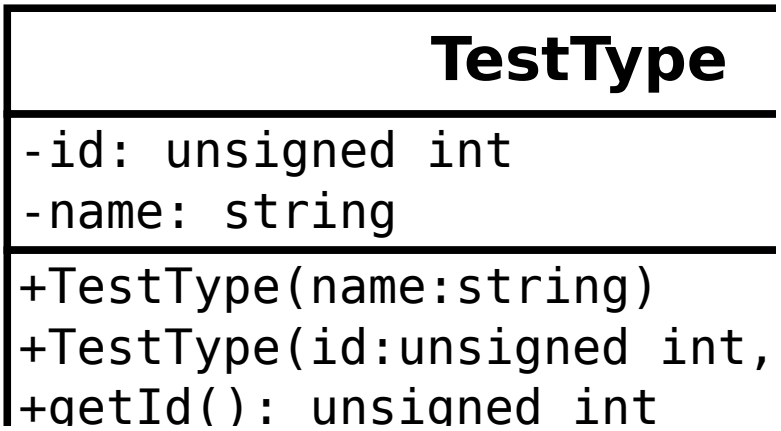

School
signed int
string
City
(name:string,city:City)
(id:unsigned int,name:string,city:City)
): unsigned int
e(): string
y(): City
e(name:string): void
y(city:City): void
Class

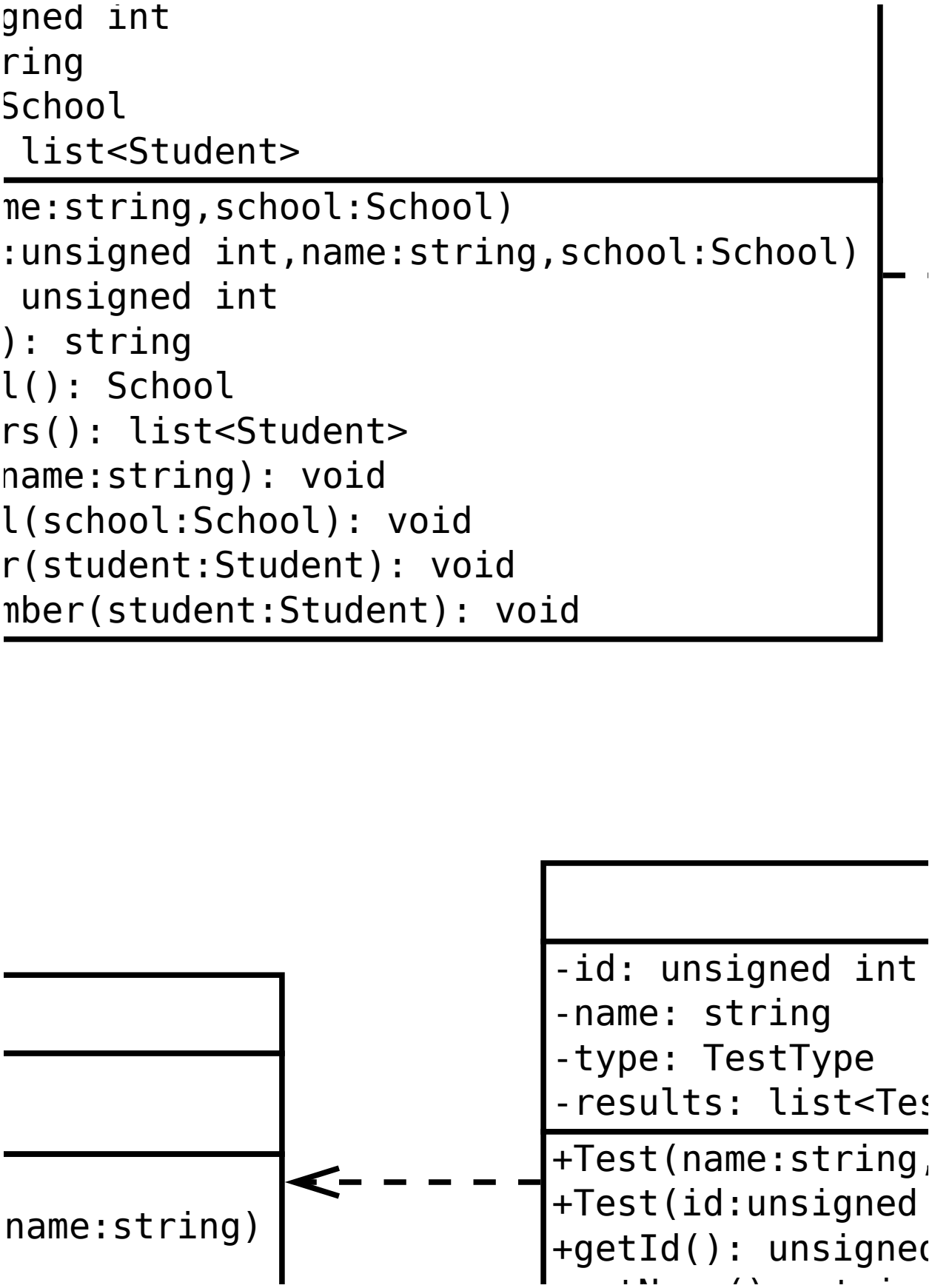

-id: unsigne
-name: strin
+City(name:s
+City(id:uns
+getId(): un
+getName():
+setName(nam

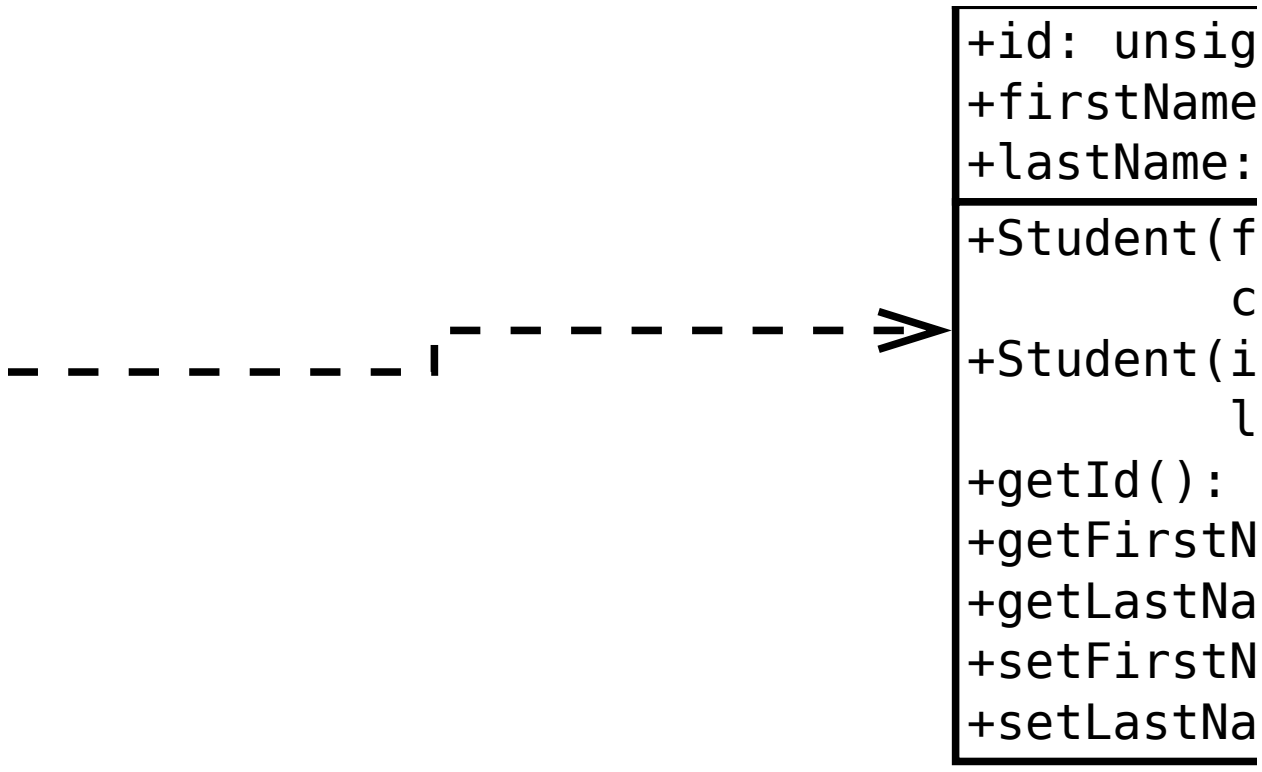

City
d int
g
tring)
igned int,name:string)
signed int
string
e:string): void

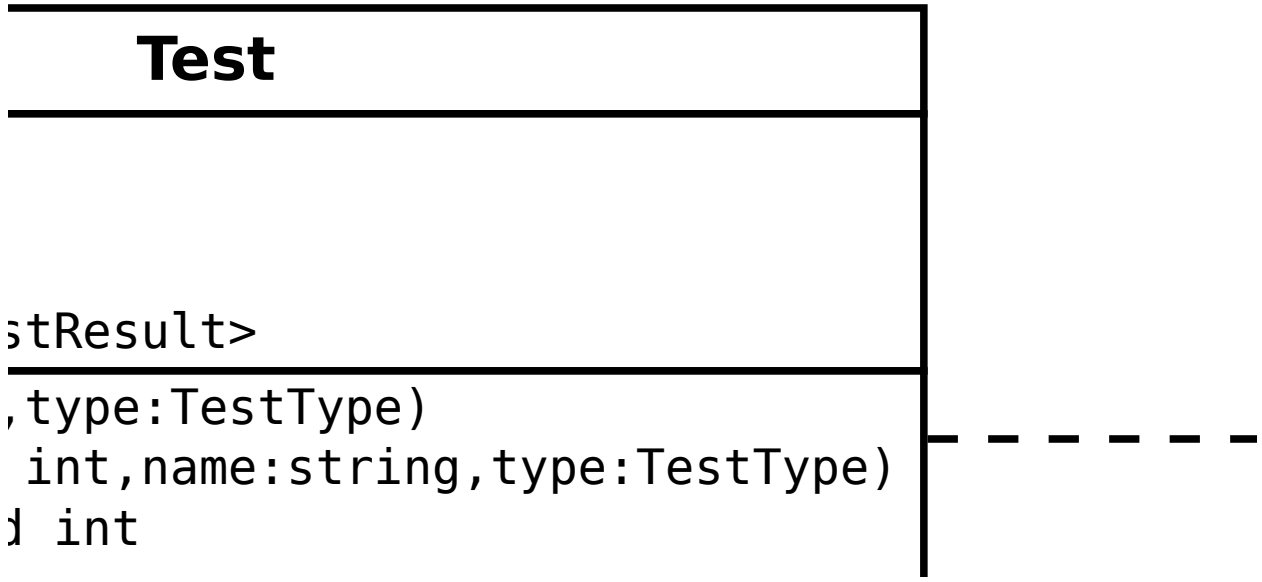

Student

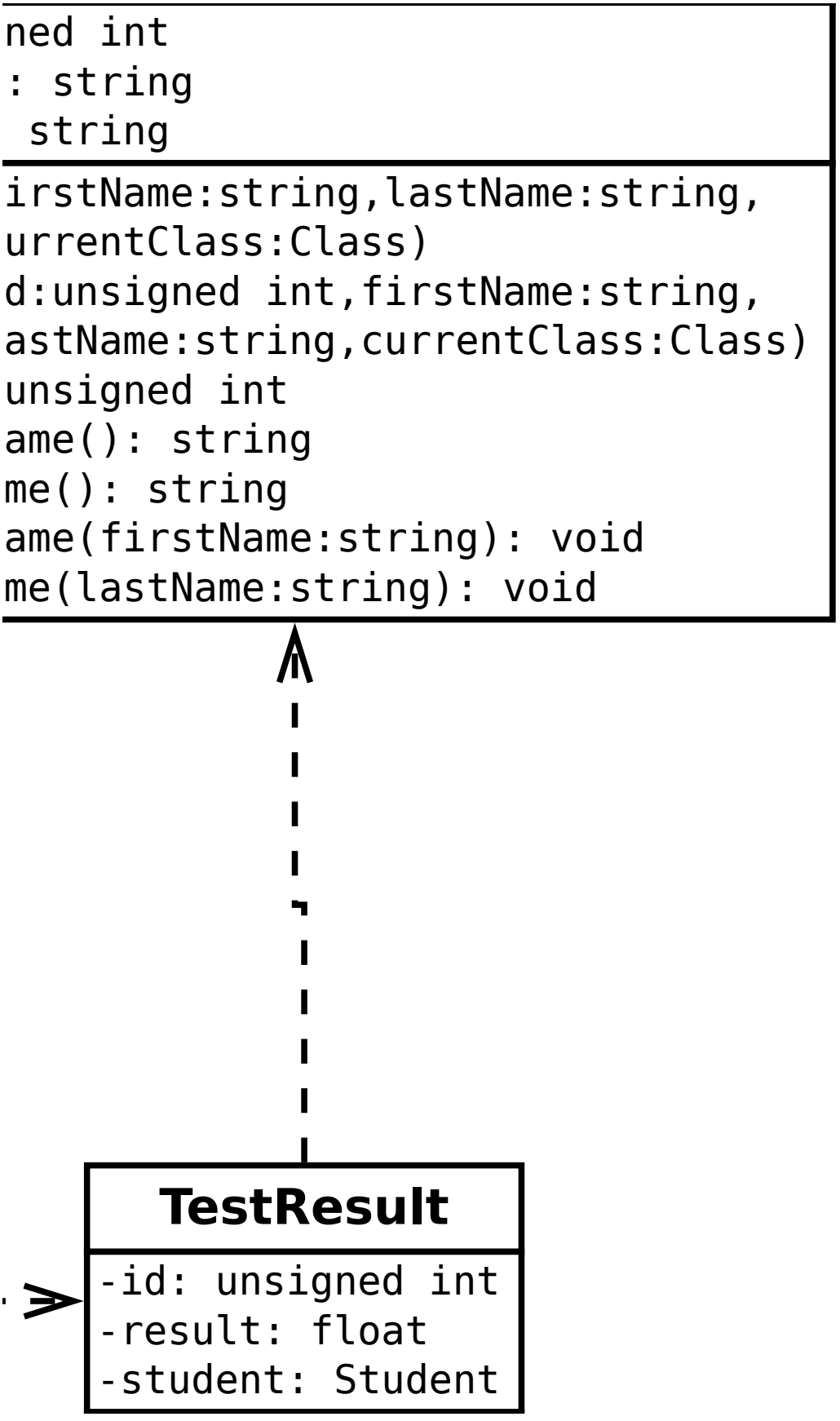

-id: unsi
-name: st
-school:
-members:
+Class(na
+Class(id
+getId():
+getName(
+getSchoo
+getMembe
+setName(
+setSchoo
+addMembe
+removeMe

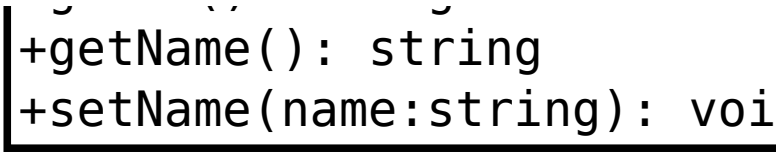

TestType
-id: unsigned int
-name: string
+TestType(name:string)
+TestType(id:unsigned int,
+getId(): unsigned int

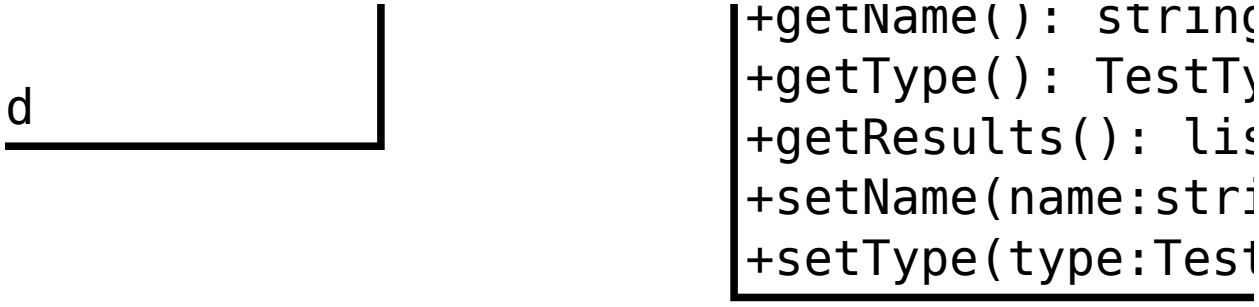

gned int
ring
School
list<Student>
me:string,school:School)
:unsigned int,name:string,school:School)
unsigned int
): string
l(): School
rs(): list<Student>
name:string): void
l(school:School): void
r(student:Student): void
mber(student:Student): void
-id: unsigned int
-name: string
-type: TestType
-results: list<Tes
+Test(name:string,
+Test(id:unsigned
+getId(): unsigned
name:string)

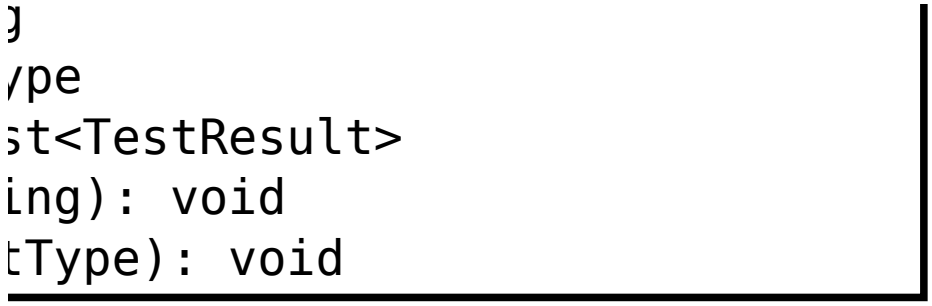

+id: unsig
+firstName
+lastName:
+Student(f
c
+Student(i
l
+getId():
+getFirstN
+getLastNa
+setFirstN
+setLastNa


Test
stResult>
,type:TestType)
int,name:string,type:TestType)
d int


ned int
: string
string
irstName:string,lastName:string,
urrentClass:Class)
d:unsigned int,firstName:string,
astName:string,currentClass:Class)
unsigned int
ame(): string
me(): string
ame(firstName:string): void
me(lastName:string): void
TestResult
-id: unsigned int
-result: float
-student: Student


+getName(): string
+setName(name:string): voi


d
+getName(): string
+getType(): TestTy
+getResults(): lis
+setName(name:stri
+setType(type:Test


g
ype
st<TestResult>
ing): void
tType): void