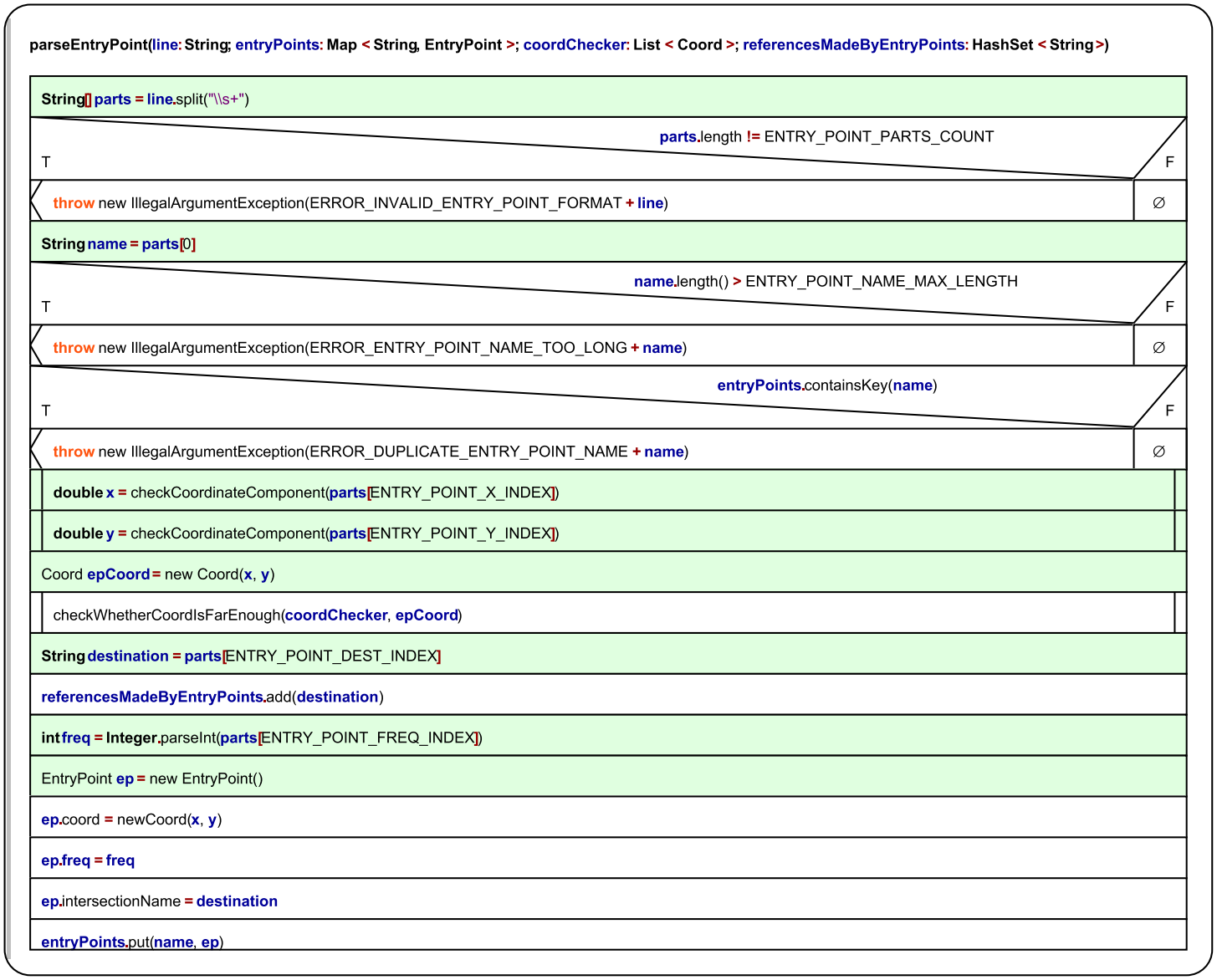
parseEntryPoint(line: String, entryPoints: Map < String, EntryPoint >, coordChecker: List < Coord >, referencesMadeByEntryPoints: HashSet < String >)
String[] parts = line.split("\\s+")
parts.length != ENTRY_POINT_PARTS_COUNT
T
F
throw new IllegalArgumentException(ERROR_INVALID_ENTRY_POINT_FORMAT + line)
∅
String name = parts[0]
name.length() > ENTRY_POINT_NAME_MAX_LENGTH
T
F
throw new IllegalArgumentException(ERROR_ENTRY_POINT_NAME_TOO_LONG + name)
∅
entryPoints.containsKey(name)
T
F
throw new IllegalArgumentException(ERROR_DUPLICATE_ENTRY_POINT_NAME + name)
∅
double x = checkCoordinateComponent(parts[ENTRY_POINT_X_INDEX])
double y = checkCoordinateComponent(parts[ENTRY_POINT_Y_INDEX])
Coord epCoord = new Coord(x, y)
checkWhetherCoordIsFarEnough(coordChecker, epCoord)
String destination = parts[ENTRY_POINT_DEST_INDEX]
referencesMadeByEntryPoints.add(destination)
int freq = Integer.parseInt(parts[ENTRY_POINT_FREQ_INDEX])
EntryPoint ep = new EntryPoint()
ep.coord = newCoord(x, y)
ep.freq = freq
ep.intersectionName = destination
entryPoints.put(name, ep)